



Universidad
Carlos III de Madrid
www.uc3m.es

Grado de Ingeniería Industrial especialidad Mecánica.

“Trabajo de final de grado”

Optimización y desarrollo del entorno visual de un programa computacional para el análisis transitorio en tuberías de flujos no adiabáticos con cambio de fases.

Autor: David Castaño Mariscal.

Tutor: Ing. Ricardo Alberto López Silva.

Leganés Junio 2015.

Título: Optimización y desarrollo del entorno visual de un programa computacional para el análisis transitorio en tuberías de flujos no adiabáticos con cambio de fase.

Autor: David Castaño Mariscal.

Director: Ricardo Alberto López Silva.

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Grado el día __ de _____ de 20__ en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de_____

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos.

Quería agradecer a todas las personas que me han ayudado en la realización de este trabajo de fin de grado y a lo largo de mis estudios, su esfuerzo y dedicación.

A toda mi familia, en especial a mis padres, Benito y Alicia, y mi hermano Juan Manuel, sin su apoyo no hubiera sido posible.

A mis amigos de Leganés, por todos los buenos ratos que me han hecho disfrutar y por apoyarme en todo momento y circunstancias.

A todos mis compañeros de universidad, sin los cuales no hubiese llegado hasta aquí, en especial a Javier Artiaga y Alberto Pillado, unos grandes compañeros de fatigas durante todo este tiempo y grandes consejeros en lo referente a los estudios y en el día a día.

A mi abuela Julia, por creer en mí y apoyarme en todo momento, la cual siempre tuvo el sueño de que su nieto obtuviera una carrera algún día.

A Inés, mi pareja, mi amiga, mi compañera en este camino. Mi fuente de energía y consejo, que siempre ha estado en los buenos y malos momentos.

Y no olvidar también agradecer a mi tutor Ricardo López, el cual siempre ha tenido momento para aconsejarme y el cual le estaré siempre agradecido del trato y la atención que me ha correspondido para realizar este proyecto.

Resumen.

En este trabajo se desarrollará una interfaz gráfica para la simulación numérica unidimensional de flujos con cambio de fase en una tubería. Dicha interfaz gráfica ha sido desarrollada en el lenguaje de programación de Matlab ^(R). En el desarrollo de código computacional se parte del código desarrollado en el artículo (1) el cual consiste en la integración numérica del modelo cuasi-homogéneo 1-D transitorio.

El programa, bautizado con el nombre **Casal** (nombre que viene de la primera parte de mi primer apellido, Castaño, y final del segundo apellido, Mariscal), permite simular la evaporación en un tubo, pudiendo acoplar un modelo de pared, variar las correlaciones para el factor de fricción (f), modelos de sub-enfriamiento y adicionalmente contiene librerías con las propiedades de los materiales más comúnmente utilizados.

Se empezará el proyecto haciendo una referencia a las partes de una interface gráfica y su importancia en líneas generales, para luego especificar algunas curiosidades del programa realizado y acabar con una muestra del programa con comentarios e ideas referentes a él y su desarrollo.

Contenido

1. Introducción.	1
1.1. Motivación.	2
1.2. Objetivos y alcances.	2
2. Flujos bifásicos. Conceptos teóricos importantes.	4
2.1. Dimensionamiento de tuberías de procesos en flujo bifásicos.	4
2.2. Fracción de vacío.	6
2.3. Modelo cuasi-homogéneo 1-D.....	7
2.4. Ecuaciones de cierre.	9
3. Simulación numérica.....	11
3.1. Errores.....	11
3.2. Discretización espacial y temporal.	12
3.2.1. Discretización espacial.	12
3.2.2. Método de Runge-Kutta y discretización temporal.	14
3.3. Ejemplos de programa de simulación numérica.	15
4. Matlab y GUI.	16
4.1. Matlab.	17
4.2. MATLAB-GUI	17
4.2.1. Flujo de trabajo en GUI.	18
4.2.2. Elementos principales utilizados de GUI.	20
4.2.3. Funciones más importantes.	21
4.2.4. Generación de ejecutables.	22
5. Desarrollo Casal.	24
5.1. Cómo distribuir parámetros de entrada.....	24
5.2. Mensajes de alarma o fallo.	25
5.3. Utilizacion de excel.	26
5.4. Evolución de guardado de salida.	26
5.5. Ultima actualizacion, graficas.....	26
5.6. Problemas.....	27

6.	Casal.....	28
6.1.	Arquitectura de la interface gráfica creada.	28
6.2.	Recogida de datos (módulo 1).	30
6.2.1.	Xzinicio (inicio).....	31
6.2.2.	Xzvnum (variable numérica).	33
6.2.3.	Xpfisc (propiedades físicas del fluido).....	36
6.2.4.	Xzcborde (condiciones de borde).	38
6.2.5.	Xzpfeltubo (Propiedades físicas del tubo).....	40
6.2.6.	Xzcinicio (Condiciones de inicio).	42
6.3.	Cálculo (modulo 2).	44
6.4.	Graficar (módulo 3).	49
6.5.	Excel.	51
6.5.1.	Excel de valores de entrada.	51
6.5.2.	Excel de valores de salida.....	52
7.	Resultados.	53
7.1.	Comparación de tiempo de cómputo con diferentes modelos de Runge-Kutta. 53	
7.2.	Diferentes ángulos de inclinación de la tubería.	60
7.3.	Diferentes flujos de calor suministrados.....	62
7.4.	Pruebas sin tener en cuenta las paredes de la tubería.....	65
7.5.	Diferentes materiales de tubería.	68
8.	Conclusiones.	71
9.	Futuras líneas de trabajo y desarrollos futuros.....	72
10.	Presupuesto.....	73
11.	Bibliografía.....	75

Índice de ilustraciones.

Figuras.

Figura 1 Patrones o régimen de flujo. _____	5
Figura 2 Señal que se esperaría de una sonda colocada frontalmente en flujo bifásico. _____	6
Figura 3 Mallas con nodos centrados en los elementos y centrado en los vértices. _____	13
Figura 4 Flujo de funcionamiento de una GUI. _____	19
Figura 5 Tipos de archivos ejecutables desde Matlab. _____	22
Figura 6 Diagrama del programa Casal. _____	28
Figura 7 Diagrama de valores de entrada. _____	30
Figura 8 GUI xzinicio. _____	31
Figura 9 GUI xznum. _____	33
Figura 10 GUI xpfisc. _____	36
Figura 11 Ventana de error. _____	37
Figura 12 GUI xzcborde _____	38
Figura 13 GUI xzpf deltubo. _____	40
Figura 14 GUI xzcinicio. _____	42
Figura 15 GUI xzoperaciones. _____	44
Figura 16 GUI xzgraficar. _____	49
Figura 17 Excel de entrada de valores.Libro1.xlsx. _____	51
Figura 18 Excel de resultados. base2.xlsx _____	52

Gráficas.

Gráfica 1 Transferencia de calor sufrida por el fluido, a 300 Kw/m ² .	55
Gráfica 2 Fracción de vacío a 300 Kw/m ² .	56
Gráfica 3 Presión a 300Kw/m ²	56
Gráfica 4 Transferencia de calor sufrida por el fluido, a 400 Kw/m ² .	57
Gráfica 5 Fracción de vacío a 400 Kw/m ² .	58
Gráfica 6 Presión a 400Kw/m ² .	58
Gráfica 7 Comparación de tiempos de un modelo a otro.	59
Gráfica 8 Comparación de variación de presión según ángulo de inclinación de tubería.	61
Gráfica 9 Fracción de vacío, a diferentes valores de flujo de calor suministrado.	63
Gráfica 10 Presión, a diferentes valores de flujo de calor suministrados.	64
Gráfica 11 Fracción de vacío, a diferentes valores de flujo de calor suministrados sin tener en cuenta las paredes de la tubería.	66
Gráfica 12 Presión, a diferentes valores de flujo de calor suministrado sin tener en cuenta las paredes de la tubería.	67
Gráfica 13 Fracción de vacío para diferentes materiales de tubería.	68
Gráfica 14 Presión para diferentes materiales de la tubería.	69
Gráfica 15 Transferencia de calor sufrida por el fluido para diferentes materiales de la tubería.	69

Tablas

Tabla 1 Valores de entrada primera prueba. _____	54
Tabla 2 Para la segunda prueba. _____	61
Tabla 3 Prueba con diferentes flujos de calor. _____	63
Tabla 4 Sin tener en cuenta las paredes de la tubería. _____	66
Tabla 5 Diferentes valores de materiales. _____	68
Tabla 6 Presupuesto total personal. _____	74
Tabla 7 Presupuesto en material informático y licencias. _____	74
Tabla 8 Presupuesto en personal. _____	74

Nomenclatura

A_c Sección de paso (m²)

a Matriz de Butcher

D Diámetro

f Factor de fricción

G Flujo másico (kg·m⁻²·s⁻¹)

g Constante de la gravedad

h Entalpía específica (kJ·kg⁻¹)

L Longitud de la tubería

\dot{m} Flujo másico (kg·s⁻¹)

n Número de iteración

P Presión (Pa)

P_H Perímetro hidráulico (m)

q'' Flujo de calor (kW·m⁻²)

t Tiempo (s)

t_{CPU} Tiempo de cómputo

u Función genérica

x Calidad

z Coordenada axial

Griega

α Angulo (rad)

ΔV Volumen diferencial

Δt Paso de tiempo (s)

ΔZ Paso especial (m)

ε Fracción vacía

ρ Densidad (kg·m-3)

Θ Multiplicador de Lockhart y Martinelli

Sub-índices

$Ee\ e\ w$ Coordenadas

$W\ P\ E$ cardinales

TP Bifásico

nb Nodos vecinos

Super-índices

s Estación

r Contador

1. Introducción.

El estudio de flujos con cambio de fase es de gran importancia tanto en la ciencia como en la ingeniería, su importancia radica en que estos flujos están presentes en gran cantidad de equipos que conforman sistemas térmicos, por ejemplo en evaporadores y condensadores presentes en sistemas de climatización y/o refrigeración, en generadores de vapor ampliamente utilizados en sistemas de generación de potencia, en sistemas solares e incluso en sistemas nucleares, por mencionar algunos.

El empleo de técnicas de simulación numérica 1-D para el análisis de estos sistemas es muy conveniente debido a su economía, robustez, rapidez y fiabilidad en los resultados, siendo reforzada su utilización por el incremento en el poder de cálculo de los actuales procesadores de un ordenador.

Programas computacionales diseñados para este tipo de estudio existen muchos, quizás el ejemplo más representativo es el programa Relap5 ampliamente utilizado en el diseño de sistemas de seguridad de plantas nucleares, otros ejemplos pueden ser los programas Fluidflow, Flownex, etc. En ese sentido este proyecto consiste en una optimización y desarrollo de un entorno visual de un programa computacional para el análisis transitorio de tubería de flujos no adiabáticos con cambio de fases. Se desarrollará utilizando el programa Matlab, para crear la interfaz gráfica del programa Casal, el cual efectuará una simulación numérica sobre un fenómeno físico de tipo dinámico. El programa simulará la evolución temporal de un fluido dentro de una tubería sometido a un flujo de calor constante. El modelado se realiza con el modelo cuasi-homogéneo y para su aproximación numérica se ha incorporado el esquema diagonalmente implícito de Runge-Kutta (ESDIRK) aumentando así su precisión temporal.

Para el desarrollo de la interfaz gráfica se hizo uso de la herramienta GUI (graphic user interface) de Matlab, se desarrolló las rutinas de I/O de datos y visualización gráfica de las variables de interés en función del tiempo.

1.1. Motivación.

Como parte de un proyecto más ambicioso que pretende simular en estado transitorio intercambiadores de calor, mi tutor el Ing. Ricardo López Silva ha desarrollado un código como parte de su tesis doctoral en lenguaje de programación de Matlab para la simulación en estado transitorio de flujos con cambio de fase de una tubería. En ese sentido, el siguiente paso fue desarrollar el entorno gráfico para la utilización del programa sin necesidad de ejecutar el programa por medio del command window o sin necesidad de tener instalado previamente Matlab ® en un ordenador.

El programa **Casal** servirá no solo para el estudio de la evolución temporal de flujos con cambio de fase desde el punto de vista de investigación, sino posteriormente podrá ser distribuido para fines de docencia, lo cual imprime una importancia mayor pudiendo servir para la comprensión de este tipo de flujos por parte de la comunidad universitaria

1.2. Objetivos y alcances.

Objetivo general.

Desarrollar un entorno visual en lenguaje de programación de Matlab ® para la simulación transitoria de flujos con cambio de fase en una tubería. Asimismo, se pretende programar la aplicación, de tal modo que sea apta para ser estudiada y modificada en un futuro.

Objetivos Específicos.

- Desarrollar las rutinas de I/O de datos.
- Acoplar a las rutinas de código anteriormente desarrolladas distintas opciones para la evaluación de las ecuaciones de cierre.
- Realizar pruebas con el programa desarrollado y determinar el efecto entre los distintos órdenes de precisión temporal empleados en la formulación.
- Crear un programa operativo y de utilidad.

El programa se ha concebido con las siguientes premisas.

- La interfaz debe ser intuitiva y sencilla de uso, de modo que cualquier persona que se disponga a utilizarla lo pueda hacer sin dificultad, buscando que tenga un aspecto agradable y lo más limpio posible.
- El código interno del programa debe ser claro y bien estructurado, para que se puedan modificar en el futuro, ya que se intenta conseguir que pueda ser mejorado y que se le puedan añadir más ampliaciones.
- Los cálculos realizados en el programa Casal deben ser consecuentes y tener un sentido real correcto, para que pueda ser utilizado en caso de que sea requerido como herramienta para estudio de tuberías o para aprendizaje de sus usuarios.
- Los cálculos realizados deben ser precisos, fáciles de entender y que no sea de complicada asimilación para el usuario.
- Se debe intentar en medida de lo posible que el programa sea lo más compatible posible con la mayoría de versiones de Matlab y con la mayoría de resoluciones de pantalla de ordenadores.

Destacar que, desde un principio tenía clara la idea de crear el programa que pudiera ser modificado por otra persona en el futuro, si fuera necesario, y por tanto el aspecto es sencillo para su posible futura programación por más personas.

2. Flujos bifásicos. Conceptos teóricos importantes.

La presencia de flujos bifásicos es frecuente en la industria. Tales flujos se encuentran en evaporadores, generadores de vapor, etc.

Los sistemas de flujo bifásicos pueden ser: gas- líquido, gas-sólido, líquido-sólido y líquido - líquido. En este trabajo, solo se hará hincapié a mezclas líquido-gas, ya que solo utilizaremos un tipo de fluido y nunca en estado sólido.

Los factores principales que determinan el régimen de flujo bifásico en una tubería son: la diferencia de las propiedades físicas y velocidades del flujo (superficiales o de fase), diámetro, dirección (ascendente, descendente u horizontal) y orientación de la tubería (horizontal, vertical o inclinada).

El estudio numérico en flujos bifásicos contribuye de manera efectiva a la comprensión de este fenómeno, permite obtener los datos necesarios para la comprensión del fenómeno físico y en el ámbito de la ingeniería nos permite hacer desarrollos y diseños de nuevos equipos y/o desarrollar estrategias de control. En este trabajo, se realiza un estudio numérico y teórico para evaluar diferentes modelos empíricos y mecanicistas comúnmente utilizados para predecir patrones de flujo bifásico de líquido-gas en tuberías horizontales y ligeramente inclinadas, aunque la formulación empleada soporta cualquier tipo de orientación.

2.1. Dimensionamiento de tuberías de procesos en flujo bifásicos.

Es de gran importancia las dimensiones de una tubería para procesos bifásicos. La selección del diámetro de una tubería para una situación específica de proceso resulta de un balance económico. Una tubería de gran diámetro, requiere mayor inversión de dinero, pero a su vez, abarata los costos de bombeo y viceversa. Los tramos cortos de tubería de planta no proporcionan la misma posibilidad de balance económico. En este

caso, la selección se basa en experiencias anteriores de caída de presión y velocidad lineal en tuberías.

Cuando un flujo bifásico fluye simultáneamente en una tubería, su distribución de fase al cambiar las propiedades de los flujos puede adoptar diferentes configuraciones físicas o regímenes de flujo a lo largo de la tubería.

Regímenes De Flujo:

Es posible establecer una clasificación visual de regímenes de flujo para tuberías horizontales y verticales. Los factores que determinan el régimen de flujo en una tubería de proceso de flujo bifásico, son:

- Diferencia en las propiedades físicas y velocidades de flujo de las dos fases.
- Diámetro de la tubería y accesorios.
- Dirección del flujo (ascendente, descendente u horizontal) y orientación de la tubería (horizontal, vertical o inclinada).

Flujos Horizontales y Ligeramente Inclinados:

Se han establecido varios tipos de régimen de flujo para el flujo de mezclas de gas y líquido en tuberías horizontales o ligeramente inclinadas (ángulo de inclinación inferior a 5 grados).

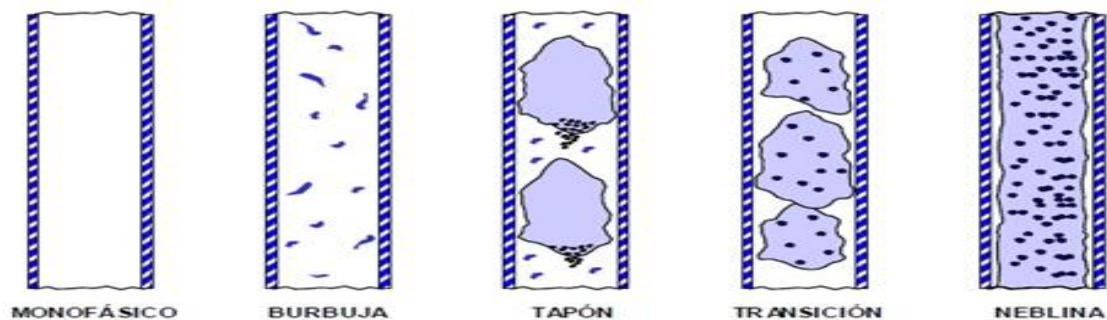


Figura 1 Patrones o régimen de flujo.

Cálculo de Caídas de Presión en Tuberías con Flujo Bifásico:

Para el cálculo de caídas de presión con flujo bifásico, es mejor realizarlo mediante software de diseño, ya que por la complejidad de tener dos fases presentes a la vez en una misma tubería existen mayor número de incertidumbres. Estos problemas se acentúan cuando tenemos situaciones donde la velocidad del fluido es inusualmente alta o demasiado bajas, o en casos en los que las propiedades del fluido son insuficientemente conocidas.

(2)

2.2. Fracción de vacío.

La fracción de vacío es uno de los principales parámetros para caracterizar un flujo de dos fases. Expresa la proporción del volumen ocupado por una determinada fase respecto al volumen total.

En particular, los conductos considerados son de geometría cilíndrica y se puede suponer simetría de traslación en la dirección axial.

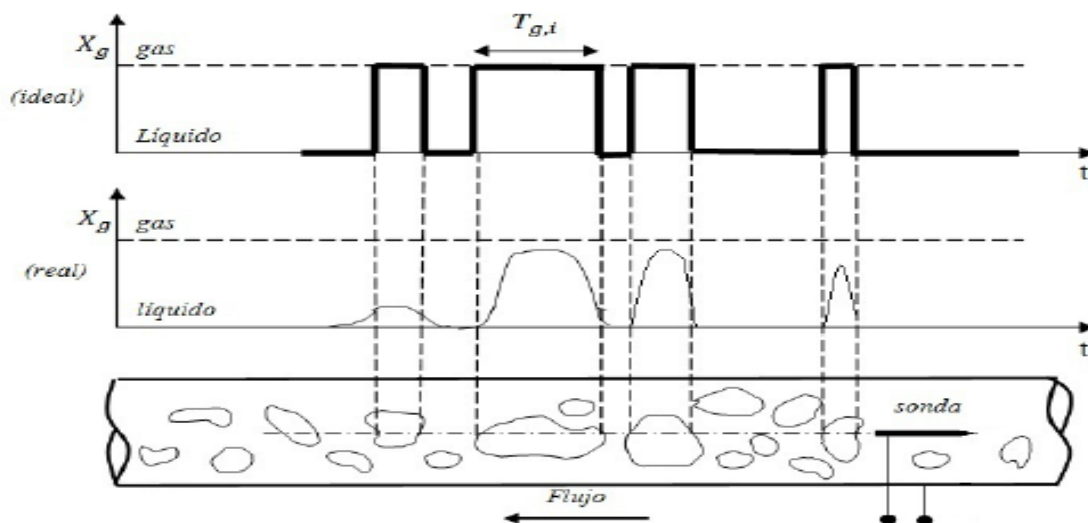


Figura 2 Señal que se esperaría de una sonda colocada frontalmente en flujo bifásico.

La fracción de vacío, por lo general varía de un lugar a otro en el canal del flujo, dependiendo del patrón del flujo de las dos fases. Se fluctúa con el tiempo y su valor es por lo general un promedio de tiempo. El caudal está relacionado con las tasas de flujo volumétrico del gas y la fase líquido y a la relación de la velocidad de las dos fases.

(3).

2.3. Modelo cuasi-homogéneo 1-D.

En este modelo se asume que el flujo es 1-D, difusión térmica en la dirección axial es despreciable

Un modelo que asume el flujo bifásico como una mezcla homogénea, que está caracterizada por un flujo másico, una presión y una temperatura de mezcla. Estas variables son obtenidas por medio de la resolución numérica de las ecuaciones de la conservación de masa, momento y energía.

Ecuación de conservación de masa.

$$\frac{\partial(\rho_{TP}A)}{\partial t} + \frac{\partial \dot{m}}{\partial z} = 0$$

Ecuación de conservación de la energía.

$$\frac{\partial}{\partial t}(\rho_{TP}h_{TP}A) + \frac{\partial}{\partial z}(\dot{m}h_{TP}) = \langle \dot{q} \rangle'$$

Ecuación de conservación de la cantidad de movimiento.

$$\frac{\partial \dot{m}}{\partial t} + \frac{\partial}{\partial z}(K\dot{m}^2) = -\frac{\partial}{\partial z}(PA) - \rho_{TP}gA \sin \alpha - \left\langle \frac{dP}{dz} \right\rangle'_{fr}$$

Dónde:

$$K = \frac{x^2}{\rho_g \varepsilon} + \frac{(1-x)^2}{\rho_L (1-\varepsilon)}$$

Y la densidad de la mezcla se puede definir como:

$$\rho_{TP} = \varepsilon \rho_v + (1-\varepsilon) \rho_l$$

Por último, para cerrar el sistema de ecuaciones se debe establecer un conjunto de condiciones iniciales y de borde o de frontera: establecen las siguientes condiciones iniciales:

Condiciones de borde o de frontera

$$\begin{cases} \dot{m}(z=0, t) = \dot{m}_{inlet} \\ P(z=0, t) = P_{inlet} \\ T(z=0, t) = \dot{m}_{inlet} \\ x(z=0, t) = x_{inlet} \end{cases}$$

Condiciones iniciales

$$\begin{cases} \dot{m}(z, t=0) = \dot{m}_0 \\ P(z, t=0) = P_0 \\ T(z, t=0) = \dot{m}_0 \\ x(z, t=0) = x_0 \end{cases}$$

2.4. Ecuaciones de cierre.

En las ecuaciones de cierre referente a lo que vamos a analizar en este documento, se pueden separar en dos grandes grupos, en los cuales vamos a tener en cuenta el modelo de fracción de vacío que vamos a utilizar y la rutina para el cálculo de pérdidas por fricción.

Esta es la forma que se utilizará para poder cerrar la formulación de forma empírica.

La definición de la fracción de vacío lo hemos comentado en unas líneas arriba, pero tendremos la opción en el programa Casal de elegir entre el modelo cuasi homogéneo, el modelo de Ahmad 1976, Levy 1967 y por último el modelo de Saha 1974.

Donde $x_a = x - x_{NVG} \exp(x / x_{NVG} - 1)$

*Saha y Levi:

$$\varepsilon = \frac{x_a}{C_o \left(\frac{x_a \Delta \rho}{\rho_l} + \frac{\rho_g}{\rho_l} \right) + \frac{\rho_g V_{gj}}{\dot{m}}}$$

Donde para Saha $V_{gj} = 1.41 \left(\frac{\sigma g \Delta \rho}{\rho_l^2} \right)^{1/4}$ y para Levi $V_{gj} = 1.18 \left(\frac{\sigma g \Delta \rho}{\rho_l^2} \right)^{1/4}$

$$C_o = 1.13$$

*Ahmad:

$$\varepsilon = \frac{x_a}{x_a + \left(\frac{\rho_g}{\rho_l} \right) S (1 - x_a)}$$

Dónde: $S = \left(\frac{\rho_l}{\rho_g} \right)^{0.205} \left(\frac{\dot{m} D}{\mu} \right)^{-0.016}$ Es la relación de velocidades superficiales o slip ratio.

Conceptos teóricos importantes.

Para la parte del factor de fricción se ha incorporado en **Casal** varias opciones para las correlaciones del factor fricción.

Ecuación de Colebrook.

$$1/(f)^{1/2} = -2.0 \log((e/D)/3.7 + 2.51/Re(f)^{1/2})$$

Moody.

$$f = 5.5 \times 10^{-3} (1 + (2 \times 10^4 e/D + 10^6 / Re)^{1/3})$$

Wood.

$$f = 0.094(e/D)^{0.225} + 0.53(e/D) + 88(e/D)^{0.44} \times Re^a$$

(8)

3. Simulación numérica.

La simulación numérica, se ha convertido en una herramienta ampliamente utilizada para el análisis y diseño de sistemas y equipos. Su objetivo es determinar las consecuencias según la formulación teórica y poder estudiar casos, sin tener los costes y peligros necesarios que conllevan su manufacturación que podrían acarrear si cada ejemplo fuera construido y probado.

La importancia de la simulación en grandes rasgos es que se pueden ver variaciones en el comportamiento del sistema, se puede llegar a un mejor entendimiento del sistema gracias a que se ve de forma gráfica y por datos los cambios que se producen al variar los parámetros de entrada y se puede experimentar con nuevas situaciones, de acuerdo a una formulación teórica.

3.1. Errores.

Un aspecto importante en la simulación numérica es el concerniente a los errores en la aproximada, los cuales pueden ser generados de distintas fuentes. Intentaremos englobar los más comunes en las siguientes líneas.

Posibles fuentes de errores en el método de volúmenes finitos:

Errores de modelización:

- En la modelización o linealización del problema.
- Modelización en las condiciones de contorno.
- Propiedades del objeto a modelizar.

Errores de cómputo:

- Error en la resolución interna del ordenador de los números reales, y por errores de redondeo.

Errores en la discretización:

- Difusión numérica, originados por la densidad de y del tipo de malla, dependiendo del tipo de aproximación empleada la malla puede introducir errores.
- Errores en la discretización, relacionados con el tamaño del elemento y la función de forma de los corrimientos de los nodos. Como norma general, se emplean elementos pequeños en las zonas de variación rápida de la solución y elementos grandes sean las zonas de variación lenta.

(9).

3.2. Discretización espacial y temporal.

Este proceso es el primero que se realiza normalmente para la evaluación e implementación numérica en los procesos simulados por ordenador. En el programa **Casal** es una parte fundamental en el cálculo de la tubería.

3.2.1. Discretización espacial.

Existen muchos métodos para la discretización del problema. Podemos clasificar los distintos esquemas de discretización en tres categorías principales: las diferencias finitas, volúmenes finitos y elementos finitos. Todos estos métodos requieren una previa

discretización geométrica (espacial) para poder realizar la discretización de las ecuaciones que gobiernan el fluido.

El tipo de discretización espacial a emplear depende del tipo de discretización de las ecuaciones empleado, así como de las estructuras internas de datos empleadas para resolver el flujo. El método de las diferencias finitas requiere una malla estructurada de puntos en los que se guarda las variables del flujo. Por otro lado, en el método de los volúmenes finitos es necesario definir puntos de control en cada volumen generado. Precisamente, es en esos vértices donde se guardan las variables, que dependiendo del método que se emplee pueden ser definidos en el centro del elemento (cell-centred) o en los vértices (cell vertex) de las celdas.

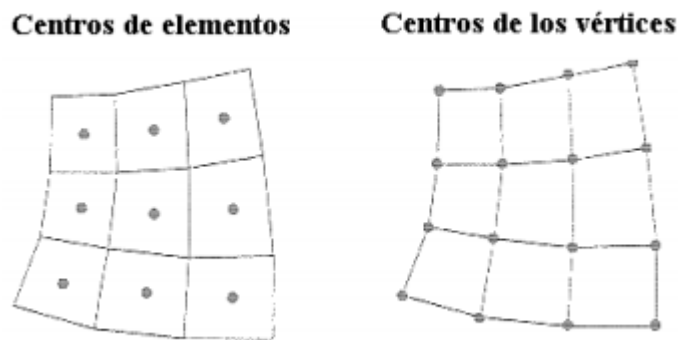


Figura 3 Mallas con nodos centrados en los elementos y centrado en los vértices.

Los métodos de volúmenes finitos, son en el presente ampliamente utilizados en los análisis ingenieriles y se puede esperar un gran incremento del uso de los mismos en el futuro, debido al carácter conservativo que éste utiliza, sobre todo en el análisis de problemas de transferencia de calor y fluidos.

La base y sentido de los volúmenes finitos, es debido a que se transforman las ecuaciones diferenciales que describen algún fenómeno físico a un conjunto de ecuaciones algebraicas y se resuelven gracias a la ayuda de un ordenador que las procesa y las calcula.

En este método se suele partir de una discretización previa del dominio de cálculo en elementos, normalmente triángulos o cuadriláteros, a partir de los cuales se construye la nueva malla de celdas o volúmenes finitos. En cada uno de estos volúmenes se realiza la discretización de la forma integral de las ecuaciones o diferencial para lo cual se hace uso del teorema de la divergencia. Las expresiones resultantes establecen la exacta conservación de propiedades relevantes del flujo en cada celda. Los términos de las ecuaciones se sustituyen por aproximaciones del tipo diferencias finitas, obteniendo ecuaciones algebraicas que se resuelven por un proceso iterativo.

Para la discretización espacial de este trabajo se basará en una discretización donde la malla está desplazada, ya que en el centro de las celdas se colocan las variables escalares, como sería el caso de la entalpía, densidad, calidad de vapor, y en las fronteras se colocarán la velocidad o flujo másico en este caso concretamente.

3.2.2. Método de Runge-Kutta y discretización temporal.

Los principales métodos de resolución de las ecuaciones transitorias se basan en esquemas de resolución independientes para el espacio y tiempo.

Los métodos de Runge-Kutta (RK) son un conjunto de métodos (implícitos y explícitos) para la aproximación de soluciones de ecuaciones diferenciales ordinarias, concretamente, del problema de valor inicial. Es un método que surge como una mejora del método de Euler. No es sólo un único método sino una importante familia de métodos iterativos tanto implícitos como explícitos para aproximar las soluciones de ecuaciones diferenciales ordinarias.

Utilización del programa.

Los esquemas de Runge-Kutta considerados en este trabajo son:

- El esquema clásico de primer orden (Euler Backward)
- Los esquemas de tercer orden de precisión de Kennedy & Carpenter (ES3), Alexander (denominado ES33a)
- El esquema de cuarto orden de Svortsov (denominado ES54).

3.3. Ejemplos de programa de simulación numérica.

Se disponen de un gran número de programas para la simulación numérica, de entre ellos vamos a destacar dos de los cuales son de gran utilidad dentro del campo del estudio de los fluidos.

- Relap5: Programa desarrollado en el Idaho National Laboratory (INL) para la U.S. Nuclear Regulation Comision (NCR). Su uso es extensivo como herramienta de análisis en diseño de plantas nucleares.

En principio es un código genérico que se puede utilizar para la simulación de una gran variedad de procesos hidráulicos y térmicos. Dispone de muchos componentes genéricos, con los cuales se pueden simular sistemas muy complejos, incluye bombas, válvulas, tubos, estructuras irradiadoras y de absorción de calor, cinética del reactor, calentadores eléctricos, bombas, jet, turbinas, separadores de vapor, acumuladores y sistemas de control.

Dispone además de modelos de procesos especiales como son el caso de lograr pérdidas de flujo en un cambio abrupto de área, ramificaciones, estrangulación de flujo y transporte de gases no condensables.

La entrada de los modelos se realiza mediante un formato de texto y los códigos incluyen la detención de errores en la entrada como ayuda a los usuarios.

- Ansys fluent: Contiene las amplias características de modelado físico que se necesitan para modelar flujos, turbulencias, transferencia de calor y reacciones para aplicaciones industriales. Utilizan modelos especiales que le otorgan la capacidad de modelar sistemas de combustión dentro de cilindros, modelos Aero acústicos, turbo maquinaria y modelos multifase.

4. Matlab y GUI.

Matlab es un potente software matemático de programación, capaz de realizar cálculos numéricos de vectores y matrices, además permite trabajar con números complejos, reales y cadenas de caracteres, así como estructuras de información más complejas. También puede realizar gráficos en 2D y 3D, convirtiéndolo en una herramienta muy avanzada para el cálculo científico técnico.

Matlab dispone de un código básico y de varias librerías especializadas (toolboxes), además de poseer una herramienta GUIDE (Graphical User Interface Development Environment) que permite al usuario la creación de interfaces gráficas, la cual posee las mismas características básicas que Visual Basic.

Entre sus herramientas y su comando help , convierte el programa Matlab en un entorno de diseño y trabajo gráfico muy intuitivo y fácil de manejar, recomendando para usuarios con pocos conocimientos en programación en general.

Algunas de sus características principales que hace que sea adecuado para el desarrollo de soluciones a problemas matemáticos y creación de interfaces graficas son:

- Gráfico para visualizar y analizar los datos de cualquiera área de trabajo.
- Lenguaje y entorno de programación interactivos
- Herramientas para construir GUIs
- Integración con aplicaciones externas.
- Posibilidad de importar datos desde archivos y dispositivos externos y de usar archivos E/S de bajo nivel.
- Conversión de aplicaciones mediante su compilador.

Todo esto hace que la herramienta Matlab este creada para que los usuarios puedan utilizarla de una manera intuitiva y con el objetivo de que su computación sea interactiva y automatizada.

4.1. Matlab.

Matlab consiste en entorno de ventanas con cuatro partes diferenciadas:

- Command Window; Ventana en la que se escriben las instrucciones que se quieren ejecutar, como operaciones matemáticas o llamadas de funciones o archivos .m.
- Current Directory: Muestra el contenido de la carpeta de trabajo. La dirección de la carpeta de trabajo.
- Workspace: Muestra información sobre las variables y objetos definidos a lo largo del desarrollo del programa.
- Command History: Es una ventana que muestra los últimos comandos ejecutados.

Matlab tiene un lenguaje de programación que dispone de sentencias para realizar bifurcaciones y bucles. Las bifurcaciones permiten realizar una u otra operación según cumpla o no una determinada condición establecida al principio. Los bucles permiten repetir las mismas operaciones o cálculos sobre datos distintos hasta que se cumpla o no una condición.

4.2. MATLAB-GUI

GUI se trata de un conjunto de herramientas diseñadas para la creación de interfaces gráficas de usuario, lo cual nos da la posibilidad de interactuar con el programa o el sistema operativo de un ordenador, para poder realizar y ejecutar programas que se necesiten ingreso continuo de datos.

Primero, es necesario entender que al crear una GUI, MatLab genera a su vez dos archivos: un archivo *.fig y un archivo *.m. Una GUI necesita ambos para poder ejecutarse y estos deben ser coherentes entre ellos.

El archivo *.fig, contiene el diseño gráfico de la GUI. Al ejecutarse por defecto, muestra el entorno gráfico final. Si se ejecuta desde el editor, permite la edición de los objetos y modificar los formatos y parámetros de los mismos. Las modificaciones que se hagan en este editor se extrapolan automáticamente al archivo *.m que contiene el código.

El archivo *.m contiene el código asociado a la figura en lenguaje M. También se denomina m-file. MatLab interpreta este código para mostrar la figura asociada, que es con la que el usuario puede interactuar. Todas las funciones se programan en este archivo.

Resulta de gran utilidad la herramienta callback a la hora de diseñar una GUI. Desde el editor GUIDE de la figura, al hacer clic derecho sobre un objeto, podemos acceder a la función del código del mismo a través de su callback. MatLab abre el archivo *.m asociado y además nos sitúa en la línea de código correspondiente. El callback, es el conjunto de líneas de código que se ejecutan al activarse un objeto.

Al crear desde cero una GUI, el m-file generado por MatLab ya contiene un código inicial.

El código tiene una estructura segmentada, ya que cada objeto de la GUI tiene unas líneas de código asociadas. A diferencia de un código normal, el código de este m-file no se ejecuta secuencialmente desde la primera hasta la última línea, sino que se ejecuta cada segmento asociado a la figura, según el usuario active cada objeto.

4.2.1. Flujo de trabajo en GUI.

El flujo de trabajo está controlado por las acciones que se llevan a cabo en el propio entorno de trabajo. Esto es debido a que en la ventana de comandos, el código y variables son fijos y a menos que se cambien antes de la ejecución del mismo, estas no cambiarán. Mientras que en la interface gráfica GUI, bajo un código preestablecido por el usuario, varía los valores de entrada de datos, así como como diferentes tipos de cálculo, desarrollo.....

Cuando se interactúa con un control, el programa registra el valor de esa opción y ejecuta los comandos prescritos en la cadena de invocación. Los menús de la interface con el usuario, los botones, los menús desplegables, los controladores deslizantes y el texto editable son dispositivos que controlan las operaciones del software. Al completarse la ejecución de las instrucciones de la cadena de invocación, el control vuelve a la interfaz para que se puedan elegir otra opción de menú. Este ciclo se repite hasta que se cierra la GUI.

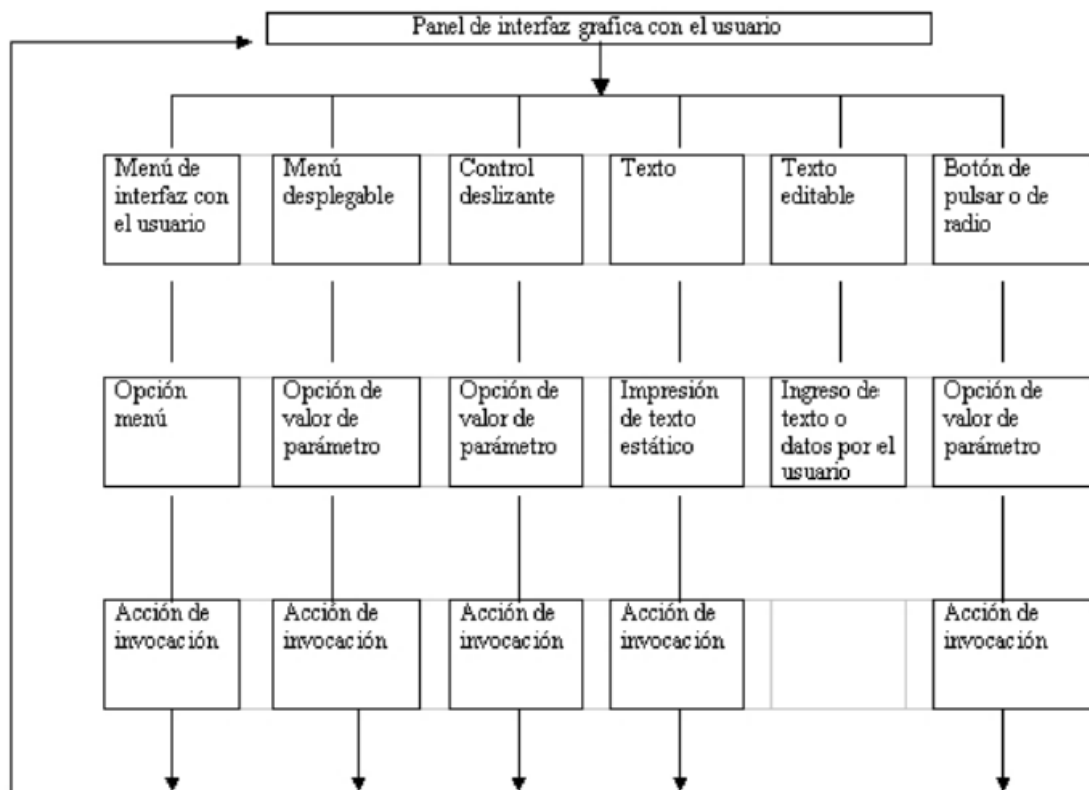


Figura 4 Flujo de funcionamiento de una GUI.

El control guarda un *String* que describe la acción a realizar cuando se invoca. Puede consistir en un solo comando de Matlab, en una secuencia de comandos o en una llamada a una función.

Los comandos con los que se trabaja son el set y get que dan y toman los valores respectivamente de las propiedades de los elementos de la GUI. Cuando se empleen, se ha de indicar no solo la propiedad con la que se está interaccionando, sino también el elemento en sí.

Todos los objetos de una GUI tienen diferentes propiedades (color, valor, posición, String, etc.) que los identifican. Todas estas propiedades se almacenan en una estructura interna a cada objeto llamada Handles. Se puede considerar que el Handles es la memoria que almacena todas las propiedades de los objetos en una GUI.

Esta memoria, al ser característica de los objetos de una GUI en concreto, se pierde al cerrar la misma. Por eso, si queremos utilizar los valores de ciertas propiedades de un objeto de una GUI en otra, tendremos que indicarlo, almacenando este valor en la memoria genérica de Matlab.

4.2.2. Elementos principales utilizados de GUI.

Se describirán los elementos que hemos utilizado para el desarrollo de la interfaz gráfica, hay muchos más elementos, pero solo se describen los que han sido de utilidad para la creación de **Casal**:

- **Push Button:** Botón rectangular, que muestra un texto en todos los casos del programa **Casal**, al cual se le asociarán acciones al hacer clic sobre él. Se ha utilizado para realizar las acciones de conexión entre ventanas de diferentes GUIs, guardado de datos mostrados en pantalla, abrir los Excels que se emplean en el programa y para ejecutar acciones de importancia en él.
- **Edit Text:** Texto editable. Se utiliza la conexión entre el usuario y el programa, ya que a través de su uso se pueden introducir una cadena de caracteres, en este caso todos numéricos.
- **Static Text:** Texto estático. Permite mostrar en pantalla un texto no modificable por el usuario. Se utiliza para que el usuario sepa información relevante sobre acciones del programa o para añadir información a las ventanas de GUI.
- **Pop-up Menú:** Crea un menú desplegable en la figura, el cual al hacer clic sobre dicho menú mostrará varios textos. Desarrollado de arriba hacia abajo, corresponde a una lista de opciones seleccionables por el usuario.

- **Axes:** Crea un objeto con ejes coordenados sobre el cual se pueden mostrar gráficas o imágenes. En el programa **Casal** lo emplearemos para las dos opciones.
- **Panel:** Agrupar objetos relacionados en la GUI.
- **Button Group:** Agrupa objetos como en el caso anterior, pero además permite establecer exclusividad mutua entre botones, dejando solamente una selección posible aunque tengamos muchas variables disponibles a la vez.

4.2.3. Funciones más importantes.

En este apartado se comentan las funciones más importantes usadas en la programación en GUI del programa en cuestión.

- **Close:** Esta función cierra la GUI especificada en la sentencia. Si en la siguiente línea se escribe el nombre de otra GUI, se consigue cerrar la presente ventana y abrir una nueva.
- **Str2double:** Función “String-to-double”. Esta función convierte un formato tipo string (código ASCII) a un formato numérico de doble precisión. Con esta sentencia se convierten los valores de entrada introducidos por el usuario dentro del programa.
- **Get y set:** Estas dos funciones nos permiten manejar datos entre objetos. Con el comando get obtenemos el valor que especifiquemos de un objeto a través del handles. Con la sentencia set establecemos el valor que especifiquemos en el handles de un objeto.
- **Global:** Declara una variable como global, de modo que quede guardada en la memoria de Matlab en vez de en el handles de la GUI. Esto nos permite

almacenar el valor de una variable y recuperarla cuando deseemos aunque cerremos la GUI. Es necesario declarar las variables globales en la GUI donde se introducirán los valores como en la GUI donde se recuperaran para su utilización.

- **Isnan:** Función que da resultado verdadero (true) si el valor que analiza es NaN (Not-a-number, no es un número) y falso (false) si el valor es un número. Se utiliza para saber si se introducen valores alfanuméricos o letras.
- **Errordlg:** Crea y abre un cuadro de diálogo de error. Con esta función podemos mostrar en pantalla mensajes de error en determinadas situaciones, limitando los valores introducidos por el usuario en el programa.

4.2.4. Generación de ejecutables.

Como hemos dichos, desde un principio, el programa **Casal** puede tener un destino instructivo y por tanto para conseguir una mayor difusión se deberá crear en forma de ejecutable. Para que se pueda utilizar en ordenadores que no tenga instalado el programa Matlab.

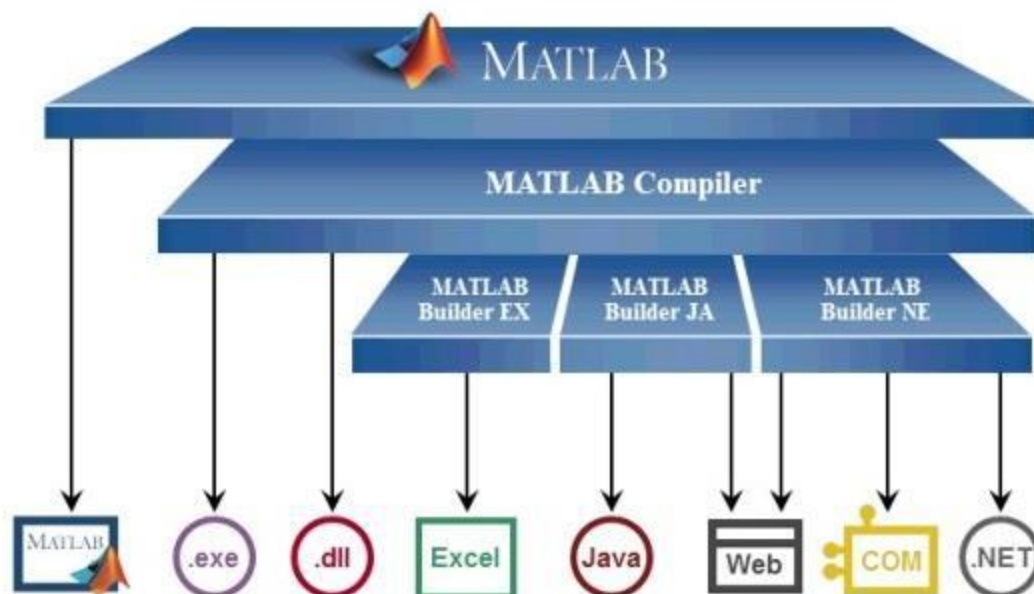


Figura 5 Tipos de archivos ejecutables desde Matlab.

Como se puede observar en la figura de encima, los archivos .exe y .dll solo se pueden ejecutar si el ordenador de destino donde se quiera ejecutar dispone del MCR de Matlab. Este MCR es Matlab Compiler, herramienta que ofrece Matlab gratuitamente.

Con este compilador se ofrece un servicio interesante, y que sin la necesidad de tener Matlab instalado, se puedan ejecutar programas creados desde Matlab.

Los pasos a seguir para crear un archivo ejecutable son:

- Introducir la instrucción `deploytool` en la ventana de comandos.
- En la ventana que aparece, se elegirá las diferentes opciones de salida de nuestro ejecutable, y podremos elegir el nombre y su futura localización.
- En la siguiente ventana se elegirán los scripts que se quieran incluir. Solo es necesario seleccionar el archivo principal (el archivo .m y .fig), ya que las subrutinas son dependientes de este archivo. Estos archivos serán incluidos en la carpeta `main function` y `other files` respectivamente.
- Para incluir el MCR tendremos que ir a la opción `packaging` y seleccionar la ventana de opción `include MCR`.
- Por ultimo solo debemos pulsar el botón de `built the proyect` y esperar a que el proceso de creación acabe, teniendo el programa ya creado en forma de ejecutable.

Algunos archivos que se utilizaran en el programa, se tendrán que llevar a la carpeta de `distr`, creada en este proceso, para que pueda ser ejecutada posteriormente.

Un claro ejemplo son las demás partes del programa, por utilizar varias m. y fig. de GUIs utilizadas, y la base de datos de los fluidos.

5. Desarrollo Casal.

En el principio se tenia claro que se queria hacer dos modulos principales del programa, una de recogida de valores de entrada y otra para integrar el programa base dentro del programa Casal. Posteriormente ,casi al final de su proceso de creacion, se introdujo la idea de utilizar los datos generados y guardados en el excel de una forma util y se creo un nuevo modulo del cual se procesaba esos datos y se generaban graficos en dos dimensiones para representar y ver mejor los valores generados.

Esto seria un pequeño resumen de su proceso de creacion de la interface grafica, ahora se concretara mas extendidamente los puntos determinantes en la creacion del programa Casal.

5.1. Cómo distribuir parámetros de entrada.

En un principio se tenia pensado crear todos los parametros que se queria poner de entrada juntos en una misma GUI para facilitar los pasos y tener todo a rapido acceso. Posteriormente se cambio de idea, se queria crear una GUI de inicio del programa y que llevara obligatoriamente a otras dos GUI para introducir los datos de entrada y de ellas a la GUI xzoperazciones donde se realizaria la simulación. En estas dos GUI seria en principio una para datos fisicos de la placa y fluidos, y la otra de valores de entrada de contorno. Se probó, en esta fase, en crear unas ventanas auxiliares (cascada) en la zona superior del programa para editar cada parametro de entrada de una forma unica.

Se desestimo la idea, de ventanas auxiliares superiores, ya que suponía un gran lío de ventanas para abrir y no era practico el paso obligatorio por todas las variables de entrada siempre. Por tanto se cambio a una GUI de inicio en la cual se pondria las principales cualidades de la tubería y luego otras GUI secundarias que se accederia a ellas atraves de push botton.

En ese momento estaba creado cinco GUI para introducir los valores de entrada, los cuales fueron separados en grupos según a que tipo de valores de entrada a los que hacia referencia. Luego se modifico y fueron cuatro solamente de acceso desde el menu de inicio, de las cual una de ellas llevaria a otra GUI especial. Esta GUI seria la especificacion de los parametros fisicos de la tuberia, que se llegaria a ella despues de seleccionar tener en cuenta las paredes de la tuberia en los calculos, ya que si no los valores no tendria sentido introducirlos para la simulación.

Esto fue en resumen los pasos que se siguieron al crear la parte de la interface grafica de introduccion de valores de entrada, una vez ya editado y finalizado añadimos muchas opciones en tipo de fluidos disponibles o diferentes tipos de factor de friccion pero ya estaba el diseño principal de las 6 GUI iniciales, la de inicio del programa y las 5 de introduccion de valores de entrada.

En este diseño siempre se ha buscado e intentado que fuera una interface grafica de acceso facil y muy intuible para una utilizacion optima de ella por cualquier usuario.

5.2. Mensajes de alarma o fallo.

En el desarrollo de la interface grafica se comprobe que cuando se introducía valores determinados el programa fallaba y no realizaba los correctamente la simulacion de calculos. Para solucionar este problema se añadió mensajes de alarma, que no dejan guardar ciertos valores en los datos de entrada para limitar estos fallos y así intentar no cometerlos. Esta idea despues tambien se aplico para evitar la introducción de letras o palabras en vez de numero como datos de entrada donde no correspondia.

Se crearon y probaron varias versiones de mensajes de alarma o fallo que se podrian generar, y cuanto mas moderna era la version de matlab utilizada, en su implementacion, mas variedad habia. Pero como estabamos utilizando matlab2012b se opto al final por uno que funcionaba correctamente y simple.

Aunque se consiguio limitar los valores en muchos de los datos de entrada utilizados aun quedaria por hacer muchas pruebas para limitarlos mas y poderlo ajustar de forma mas precisa y concreta.

5.3. Utilizacion de excel.

Las funcion global en la entrada de valores en GUI es de gran utilidad, declara una variable como global consiste en guardarla en la memoria de Matlab en vez de en el handles de la GUI, consiguiendo con eso almacenar un valor de una variable para despues poder recuperarla aunque cerremos la GUI en donde se introdujo ese valor. Se hicieron pruebas con esta funcion pero al tener tantas variables de entrada daba errores en el calculo ya que se ralentizaba la simulación del progama **Casal** . Por tanto se opto como opcion las funciones de escritura y lectura de las tablas excel , write y read, utilizando los excel como parte auxiliar del programa. Ademas destacar que es de gran utilidad grabar los resultados de salida, que consideremos relevantes, en una tabla de excel para poder ver mas concretamente lo resultados generados por el programa en su simulación.

5.4. Evolución de guardado de salida.

En un principio al guardar los datos de salida se tenia pensado grabar solamente los valores de salida de la zona elegida y su evolucion en el tiempo en el excel de resultados, para no sobrecargar el programa con variables y por los problemas que genera la utilizacion de GUI y variables. Pero al final se aplico la opcion de no guardarlo en la programacion propia de matlab, a traves de la funcion global, y la utilizacion de una tabla de excel auxiliar dando la opción de poder guardar mas datos sin complicaciones .

5.5. Ultima actualizacion, graficas.

La ultima actualizacion del programa, por motivos de tiempo ya que se pueden hacer aun mas cambio y disponemos de mas ideas para el futuro del programa , es la de introducir una parte final que seria la referida en la GUI de xzgraficar. Sirve como complemento del programa para generar las grafica de los resultados obtenidos en la simulación.

5.6. Problemas.

Destacar un problema que se ha comprobado, y es que los estados gaseosos de alta temperatura o presión alta, la simulación realizada por el programa **Casal** son más lentos y complejos, por tanto genera un aumento en el tiempo de computación del ordenador. También problemas de tiempo de computación cuando se introduce un mallado espacial de grandes dimensiones, o demasiado pequeño, provocando la ralentización del programa e incluso su detención.

Estos errores se podría solucionar de dos formas, modificando la parte de programación de Matlab en su código base y modificando las limitaciones del programa tras muchos experimentos con él, estableciendo cuales serán variables admisibles dependiendo de las circunstancias de estudio.

6. Casal.

En los siguientes apartados se comentará el entorno visual que se ha creado para este proyecto, y a continuación sus partes con los elementos importantes y destacables en cada GUI y los componentes que la forman.

6.1. Arquitectura de la interface gráfica creada.

Para entender la distribución del programa se observara en el siguiente esquema, de la estructura global del programa. Se muestra un esquema genérico, pero sirve para indicar las conexiones entre las GUI y ver los tres módulos principales.

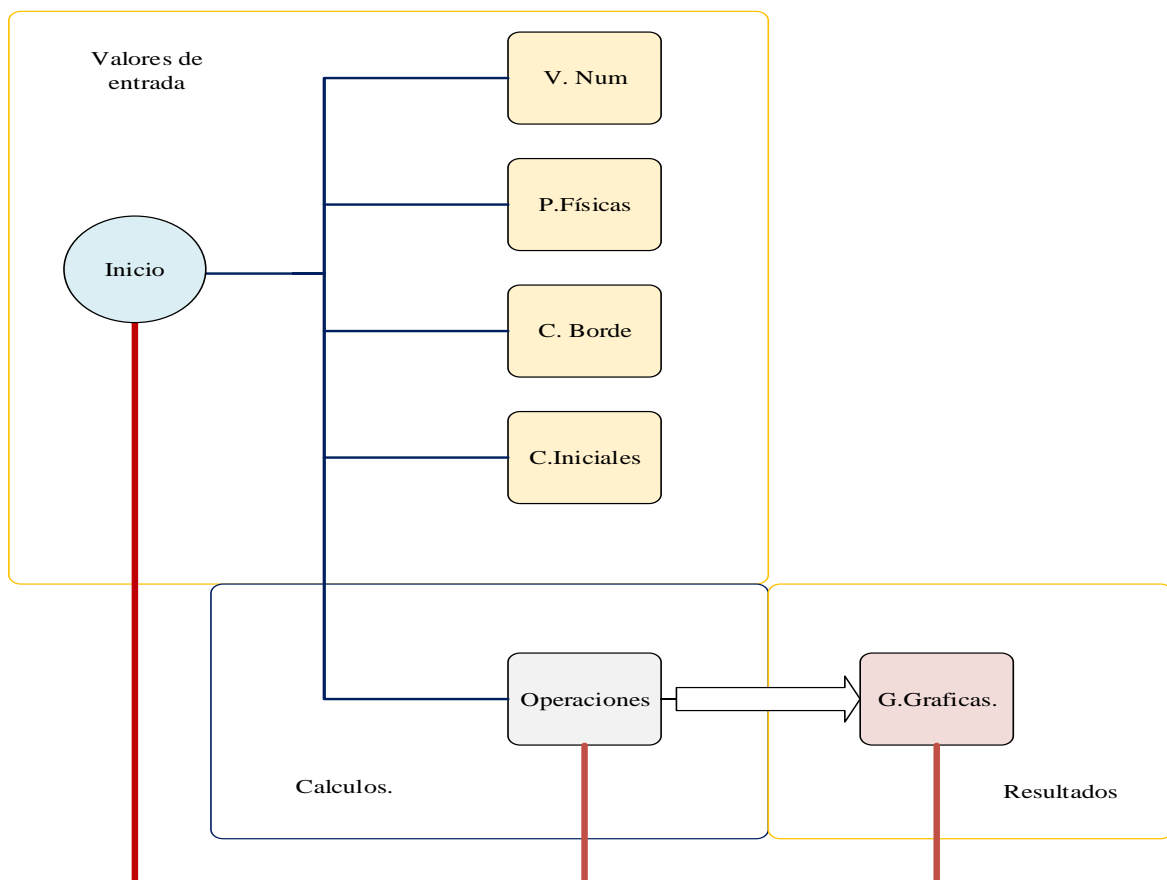


Figura 6 Diagrama del programa Casal.

La interfaz gráfica Casal, en la cual se puede comprobar en el esquema, está compuesta por 8 GUIs, las cuales están relacionadas entre sí, interactuando entre ellas.

También se puede observar una división más global, en la cual se compone de tres módulos:

- Entrada de valores o recogida de datos.
- Operaciones o cálculo.
- Gráfica de resultados o resultados.

Para facilitar su manejo tenemos acceso siempre desde cualquier GUI a menú de inicio, para poder volver modificar los valores de entrada y ver que parámetros estamos estudiando, representado con las líneas rojas en la ilustración.

Es de importancia, para entender y valorar mejor el programa, documentar independientemente cada GUI de la aplicación. Por tanto en los siguientes apartados, cada GUI, se muestra una captura de pantalla de la figura y se explica brevemente la función de cada botón de acción, cuadro de texto editable y otros elementos relevantes que disponga.

Cabe decir en este punto que en este documento no se expone la programación de la aplicación, las líneas de código de los archivos .m, ya que al confeccionarse el programa para uso interno de la UC3M, el código escrito es propiedad de la misma, mientras que el presente documento es de acceso público.

La nomenclatura que se emplea en los siguientes apartados es consistente con el nombre de cada GUI del programa.

6.2. Recogida de datos (módulo 1).

Se representa todas las GUIs de recogida de datos de una forma de diagrama para tener una mejor visión de lo que estamos tratando y los valores de entrada que podemos introducir en cada apartado.

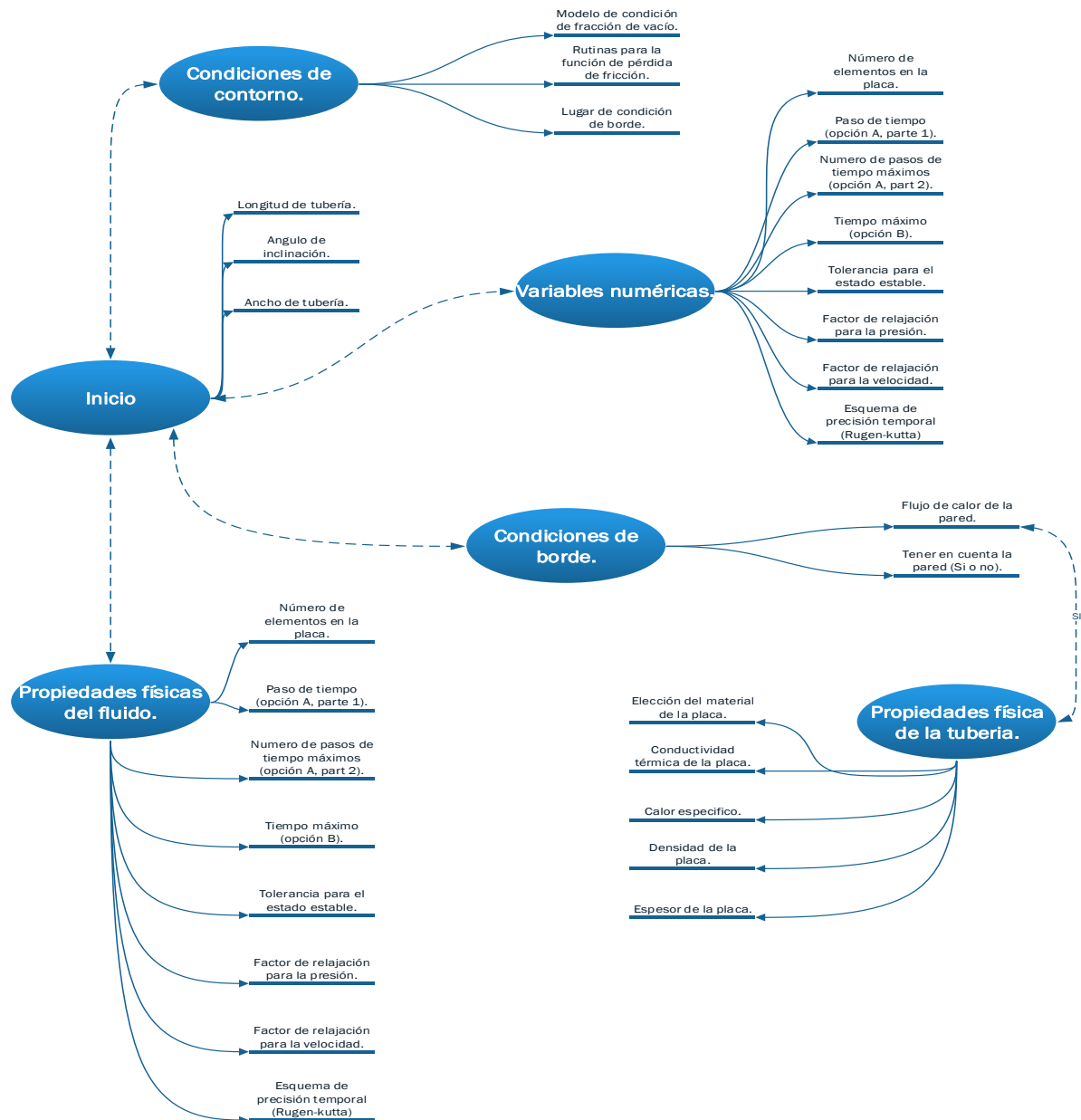


Figura 7 Diagrama de valores de entrada.

6.2.1. Xzinicio (inicio).



Figura 8 GUI xzinicio.

Se muestra el menú de inicio, con el cual se iniciara el programa.

Tenemos un título situado en la zona de arriba a la izquierda. Luego dos imágenes en la portada, la de arriba a la derecha el escudo de la universidad y abajo a la izquierda el de una tubería representativa de lo que estamos analizando.

Tenemos, como zonas importantes, dos recuadros que serían uno para la introducción de datos de entrada (parámetros del tubo) y otro (datos numéricos) que está formado por un conjunto de botones que nos llevaría a otros menús GUI para introducir más valores de entrada.

Los objetos más importantes son los siguientes:

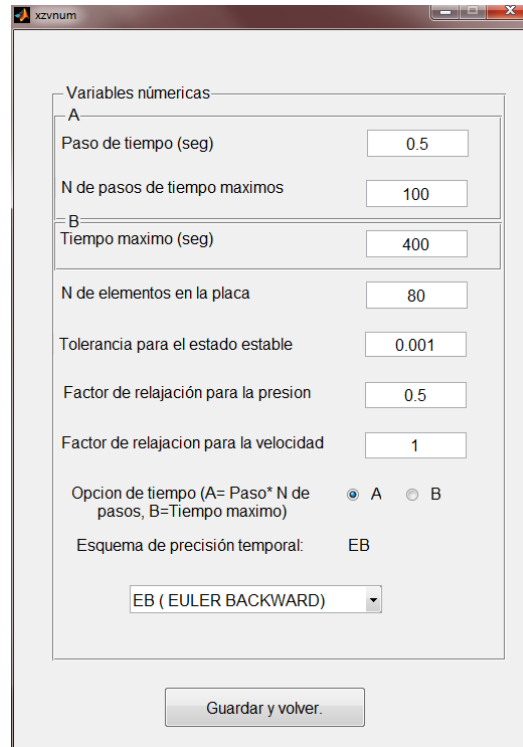
- Push buttons:
 - V numéricas: Al pulsar este botón se accede al menú de las variables numéricas.
 - Propiedades físicas: Al pulsar este botón se accede al menú de las propiedades físicas del fluido.
 - Condiciones de borde: Al pulsar este botón se accede al menú de las condiciones de borde.
 - Condiciones iniciales: Al pulsar este botón se accede al menú de las condiciones iniciales.
 - Salir: Botón para salir del programa.
 - Calcular: Botón que servirá como paso para ir del módulo 1 del programa al módulo 2, cálculo en la GUI y operaciones.

- Texto editable:

En el cuadro de parámetro de tubería, tenemos tres textos editables, en los cuales podremos añadir valores de entrada, son las principales dimensiones en la tubería.

- Longitud del canal: Tamaño de la tubería en metros.
- Ancho de la placa: Determina el diámetro de la tubería en los cálculos, ya que se considera siempre circular.
- Angulo de inclinación: Introducido en valor de grados el cual indicara el ángulo en que está colocada la tubería respecto a la horizontal de un plano.

6.2.2. Xzvnum (variable numérica).



The screenshot shows a window titled 'xzvnum' with a light gray background. It contains a section titled 'Variables numéricas' with two sub-sections, A and B. Section A includes 'Paso de tiempo (seg)' with a value of 0.5 and 'N de pasos de tiempo maximos' with a value of 100. Section B includes 'Tiempo maximo (seg)' with a value of 400. Below these are 'N de elementos en la placa' (80), 'Tolerancia para el estado estable' (0.001), 'Factor de relajación para la presion' (0.5), and 'Factor de relajacion para la velocidad' (1). There are two radio buttons for 'Opcion de tiempo (A= Paso* N de pasos, B=Tiempo maximo)', with 'A' selected. Below this is a label 'Esquema de precisión temporal: EB' and a dropdown menu showing 'EB (EULER BACKWARD)'. At the bottom is a button labeled 'Guardar y volver.'

Figura 9 GUI xznum.

En esta pantalla de GUIs se introducirá las variables numéricas del problema. Cabe destacar que existe dos botones especiales, una con la letra A y otra con la letra B, que se puede seleccionar, su misión será la de elegir una de las dos opciones de tiempo para el cálculo final.

La A será para calcular el tiempo según el paso por el número de pasos y la opción B será para introducir el tiempo máximo directamente. Si seleccionamos una no se podrá seleccionar la otra, ya que no tendría sentido tener las dos opciones elegidas a la vez o ninguna de ellas.

Los objetos más importantes son los siguientes:

- Push buttons:
 - Guardar y volver: Al pulsar este botón se guardaran los datos introducidos en esta GUI, en el Excel de valores de entrada generados (libro1.xlsx). Si no se pulsa no se procederá al guardarlo y por tanto no quedaran almacenado.
- Texto editable:
 - Paso de tiempo de tiempo: Su misión es que podamos introducir el tamaño de paso de tiempo que se aplicara en las ecuaciones del programa.
 - N de pasos de tiempo máximos: Numero de pasos de tiempo máximos.
 - Tiempo máximo: Se indica el tiempo máximo para realizar los calculos del programa, sería una forma de añadirlo sin tener que determinar el paso de tiempo ni el número de pasos totales.
 - N de elementos en la placa: Sirve para poder seleccionar en cuantos trozos del mismo tamaño realizamos el mallado de la tubería. Útil para realizar la discretización espacial que vamos a efectuar en la tubería.
 - Tolerancia Para el estado estable.
 - Factor de relajación Para la presión.
 - Factor de relajación Para la velocidad.
- Pop-up Menú: En esta GUI se da la opción de poder seleccionar que versión de discretización temporal queremos implementar en nuestros cálculos. Disponemos de 5 modelos de selección.
 - EB: (EULER BACKWARD)
 - ES3: (ESDIRK-3 BILL & CARPENTER 2002)
 - ES33a: (ES33a SKVORTSOV 2003)
 - ES33b: (ES33b R. Alexander 2003)
 - ES54: (ES54 SKVORTSOV 2003)

- Texto estático: Se destaca el texto estático, de los que hay, el que nos recuerda la selección de discretización temporal que se elegido, mostrándolo en la pantalla.

- Botón de selección (Toggle Button): En esta GUI se utilizara para seleccionar una de las dos opciones A o B.

La opción A servirá para determinar el tiempo total empleado en el cálculo del programa, a través del paso de tiempo por el número de pasos.

La opción B consistirá en introducir directamente el tiempo total del cálculo, obtenido por el tiempo máximo.

-

6.2.3. Xpfisc (propiedades físicas del fluido).

The screenshot shows a window titled 'xpfisc' with a light gray background. On the left, a box labeled 'Propiedades físicas' contains six input fields with their respective labels and values: 'Flujo másico por canal (kg/s)' with '0.109252', 'Temperatura inicial (°C)' with '225', 'Temperatura entrada (°C)' with '225', 'Presion entrada circuito (Pa)' with '1.08e+07', 'Calidad inicial circuito (Kg/Kg)' with '0', and 'Calidad de entrada circuito (Kg/Kg)' with '0'. To the right, 'Fluido del canal:' is followed by a dropdown menu showing 'AGUA'. Below this, a box labeled 'Propiedades críticas:' contains two fields: 'Temperatura critica (°C)' with '373.94' and 'Presion critica(Pa):' with '2.2064 e+0.7'. At the bottom right is a button labeled 'Guardar y salir.'.

Figura 10 GUI xpfisc.

La GUI xpfisc hace referencia a las propiedades físicas del fluido, las cuales se podrán modificar e incluso el fluido que se utilizara, dentro de la tubería.

Para propiedades del fluido, se utilizan las propiedades que vienen en el programa NIST.STANDAR.REFERENCE.Database.23. Se implementa estos valores a través de una librería obtenida del programa mencionado para el uso en **Casal**.

Para intentar evitar errores en la ejecución del programa, viene incluido un cuadro de dialogo donde a través del uso de texto estático, se añade la información al usuario de la temperatura crítica y la presión critica para cada fluido.

Cuando el usuario introduzca valores superiores, a los que vienen indicados en esa cuadro, al intentar pulsar el botón de guardar y salir saldrá un error en ventana, con mensaje. Este mensaje ira dirigido hacia el usuario, para indicar que no se puede realizar el guardado y por tanto que modifique los valores de entrada para ser admisibles a los parámetros del fluido seleccionado.

Los objetos más importantes son los siguientes:

- Push buttons:
 - Guardar y volver: botón que al pulsar se guardaran los datos introducidos en esta GUI para poder guardarlo como parámetros de entrada.

- Texto editable:
 - Flujo másico por canal: En kg/s.
 - Temperatura inicial: En C°.
 - Temperatura entrada: En C°.
 - Presión entrada circuito: Introducido en Pa.
 - Calidad inicial circuito: Valor comprendido entre 0-1.
 - Calidad de entrada de circuito: Valor comprendido entre 0-1.

- Pop-up Menú:
 - Fluido de canal: podremos seleccionar el fluido que elegir dentro de la tubería.

Los refrigerantes incorporados, una gran selección, destacan entre los más comunes el agua, NH₃, R410A, Nitrógeno, Hidrógeno, Acetona, R22..... pero la lista es muy extensa.

- Texto estático: En este apartado solo tendremos temperatura crítica y presión crítica como destacable.

La cual también se utilizara para limitar los valores que podemos poner de entrada.

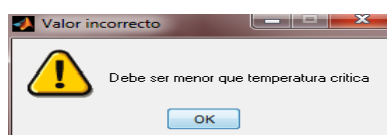


Figura 11 Ventana de error.

6.2.4. Xzcborde (condiciones de borde).

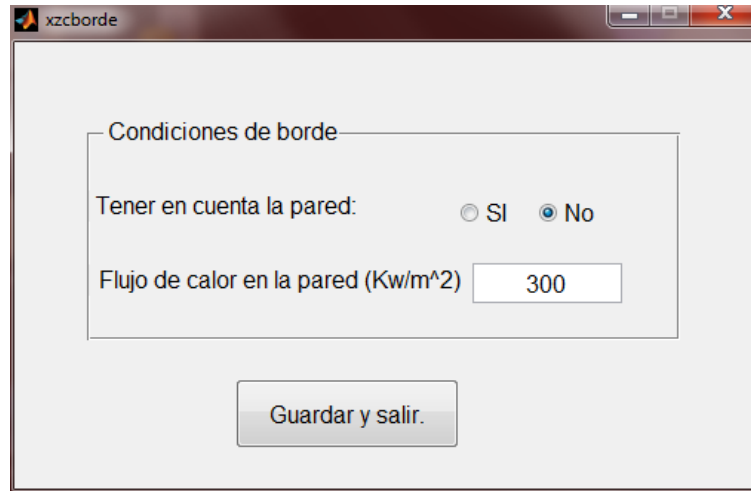


Figura 12 GUI xzcborde

Esta GUI tiene como objetivo el de introducir los valores de entrada que se utilizaran en las condiciones de borde para luego operar en el programa principal. Esta GUI es especial ya que al seleccionar el botón, SI, de tener en cuenta la pared se abrirá otra GUI para determinar las propiedades de la tubería, xzpfeltubo

Los objetos más importantes son los siguientes:

- Push buttons:
 - Guardar y volver

- Texto editable:
 - Flujo de calor en la pared: Determinaremos el flujo de calor sufrido por el fluido dentro de la tubería, en las unidades Kw/m^2 . Esto quiere decir que introduciremos la transferencia de calor, un parámetro de gran importancia en el programa y sus cálculos.

- Botón de selección: Se puede elegir no tener en cuenta la pared de la tubería facilitando por tanto los cálculos, esta era la versión del programa original, o la de tener en cuenta la pared. Considerar las paredes de la tubería conllevará una gran repercusión en la simulación realizada por el programa, pero se obtendrá valores más interesantes para su análisis.

6.2.5. Xzpfeltubo (Propiedades físicas del tubo).

The screenshot shows a window titled 'xzpfeltubo' with a standard Windows-style title bar. The main content area is titled 'Datos de la placa'. It contains a dropdown menu for 'Eleccion del material de la placa:' with 'Acero' selected. Below this is a section titled 'Otro material' which contains three input fields: 'Conductividad termica de la placa (W/mK)' with the value '372', 'Calor especifico (J/KgK)' with the value '389', and 'Densidad de la placa (Kg/m^3)' with the value '8900'. At the bottom, there is an input field for 'Espesor de la placa (m)' with the value '0.01' and a button labeled 'Guardar y salir.'.

Figura 13 GUI xzpfeltubo.

GUI dedicada a la introducción de los valores de las propiedades físicas de la tubería, indicando el material y por consiguiente sus conductividad, calor específico y densidad de la placa.

En la parte superior tenemos elección del material de la placa o tubería, en donde se puede elegir el material del que estará formada la tubería. Se puede elegir como valores de predefinidos el acero, hierro, aluminio, cobre y hormigón, considerado los materiales principales para recubrir las paredes de la tubería.

No se añadió más materiales, pero para disposición del usuario a una mayor opción de cálculos por parte del programa **Casal**, se tiene la opción de introducir manualmente, por parte del usuario, los valores que se quiera utilizar para la conductividad térmica de la placa, calor específico y densidad. Cuando seleccionamos un material los valores anteriores se modificaran y vendrán los valores de ese material por defecto.

Los objetos más importantes son los siguientes:

- Push buttons:
 - Guardar y volver:

- Texto editable:
 - Conductividad térmica de la placa: En valores de W/mK.
 - Calor específico: Introducidos en J/KgK.
 - Densidad de la placa: En Kg/m³.
 - Espesor de la placa: valor que no depende de la elección del material de la tubería sino de las dimensiones que necesitemos o requiramos para el análisis. Añadido en esta GUI ya que si no tenemos en cuenta las pared de la tubería no influiría en nada en los cálculos del programa.

- Pop-up Menú: En este menú se podrá elegir el material adecuado, dentro de unos materiales establecidos por el programa con sus propiedades por defecto:
 - Acero.
 - Hierro.
 - Aluminio.
 - Cobre
 - Hormigón.

6.2.6. Xzcinicio (Condiciones de inicio).

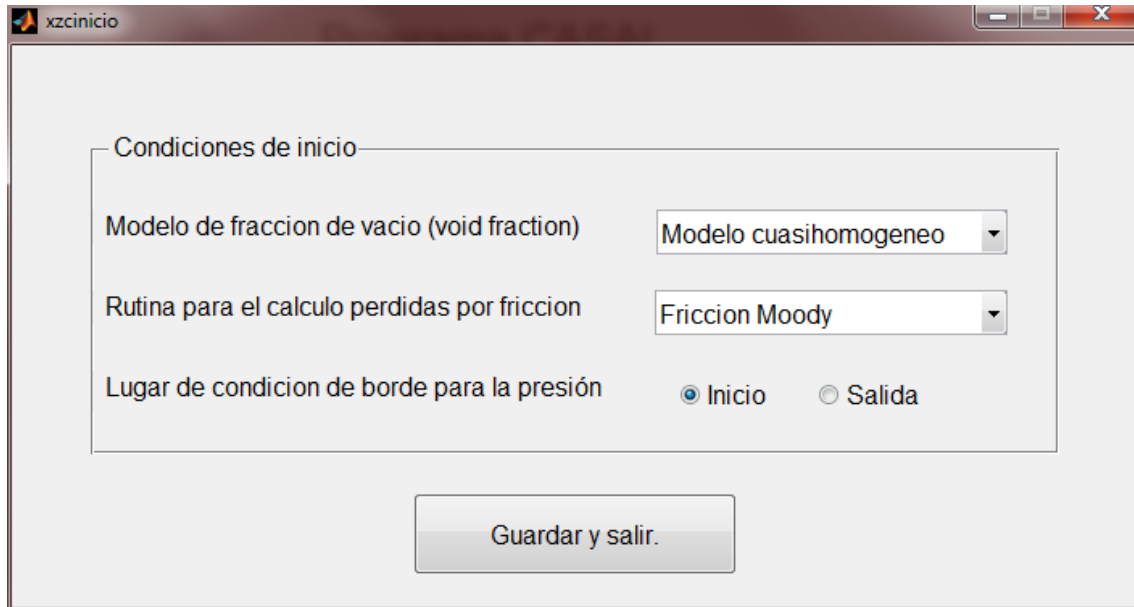


Figura 14 GUI xzcinicio.

En esta GUI tenemos las condiciones de inicio.

Separada en tres grandes partes, las cuales son la elección del modelo de fracción de vacío, la rutina para el cálculo de pérdidas por fricción y el lugar de condición de borde para la presión.

En esta ventana solo se permite elección entre varias opciones disponibles establecidas por el programa.

Los objetos más importantes son los siguientes:

- Push buttons:
 - Guardar y volver:

- Pop-up Menu: Tenemos dos menú.
 - Modelo de fracción de vacío: la cual podemos seleccionar entre modelo cuasi homogéneo, modelo de Ahmad 1970, Levy 1967 y Saha 1974
 - Rutina para el cálculo de pérdidas por fricción: Seleccionaremos que rutina utilizaremos para el cálculo de pérdidas por fricción en la tubería o en la placa por fluido a su paso. Entre estas elecciones están introducido las que se han considerado más importantes, pero no son todas:
 - Friccion Moody.
 - Friccion Darcy .
 - Friccion Serghides.
 - Friccion Colebrook.
 - Friccion Jain.
 - Friccion Churchill.
 - Friccion Wood.
 - Friccion Chen.
 - Friccion Zigrang and Sylvester.

- Botón de selección (Toggle Button):
 - Lugar de condición de borde: Se selecciona donde implementar las condiciones de borde para la presión, al principio o al final de la tubería.

6.3. Cálculo (modulo 2).

En este módulo se encuentra la parte de simulación, la cual está creada en una única GUI. Se considera la parte más importante y central del programa **Casal**. Aquí se realizarán los cálculos, ya que en él está implementado el código con las ecuaciones de fluidos que vamos a emplear, y donde se reproducirá para la obtención de los resultados.

La simulación dará comienzo cuando se pulse el botón operar, y cuando el programa acabe se guardara todos los datos de salida en una tabla de Excel. Esta tabla después se podrá guardar en el lugar que queramos y con el nombre que queramos, además de la posibilidad de generar gráficas.

Durante los procesos de simulación del programa, se muestra resultados obtenidos en las cuatro gráficas, de forma visual, que aparecen en la pantalla de la GUI. Además vendrá el valor del error, generado en todo momento, mostrado en la pantalla.

Cuando el programa acabe los cálculos, ósea sea estacionario el flujo, en pantalla se cambiara el mensaje del número de error por el de finalizado.

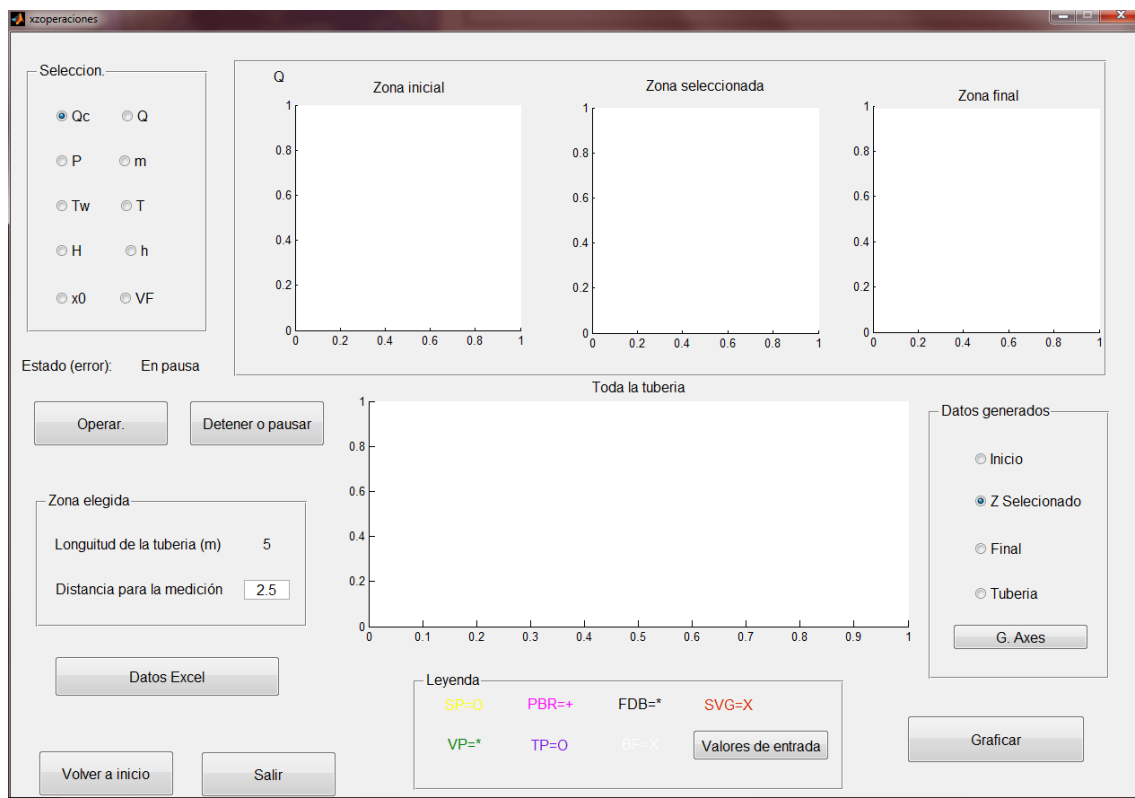


Figura 15 GUI xzoperaciones.

La grafica de abajo, justo donde está escrita toda la tubería, mostrara la evolución a lo largo de la tubería de los resultados obtenidos en cada momento en que se encuentre las simulación del programa.

Se dispone de una leyenda, abajo en el recuadro leyenda, que otorgara información importante, sobre el estado en que se encuentra el fluido dentro de la tubería, al usuario. Con colores y símbolos. Al aparecer un símbolo a la derecha, en la gráfica grande en el eje x, todo lo que quede a su izquierda estará en ese estado a no ser que encontremos otro símbolo diferente mostrado en la leyenda.

Se encuentran los principales estados que nos podemos encontrar en este estado de fluidos (leyenda):

- SP=single phase región. Región de fase simple.
- PBR=Parcial boiling región. Región parcialmente en ebullición.
- FDB=Fullydevelope boiling. Totalmente en ebullicion.
- SVG= sifnidication void generation. Generación significativa de vacío.
- VP=Fase vapor.
- TP=Segunda fase.
- Bf=two phase flow region. Región de dos fases de fluido.

Además dentro del recuadro de la leyenda se dispondrá de un botón que servirá para ver un Excel con los valores de entrada que hemos utilizado o utilizaremos en la simulación.

También tenemos un recuadro, arriba a la derecha, cuya utilidad es la de seleccionar que propiedades queremos que se represente en las cuatro ventanas de gráficas.

Luego a la derecha tendremos otro recuadro, que sirve para guardar las gráficas generadas al finalizar el proceso de operación y mostradas en la GUI.

Los objetos más importantes son los siguientes:

- Push buttons:
 - Operar: Principal botón de esta GUI donde dará comienzo la simulación del flujo en la tubería con todos los valores de entrada introducidos en todas las GUI anteriormente mencionadas. Cuando empieza el proceso, por método de prevención de errores en el cálculo, se bloquea los recuadros de selección, zona elegida y zona de guardado de gráficas.
 - Detener o pausar: Botón añadido por si no interesa lo que estamos calculando, simulación que se está reproduciendo en el programa o nos hemos equivocado al introducir algún valor de entrada, para parar el proceso y poder volver a modificar los parámetros de entrada.
 - Volver a inicio.
 - Salir.
 - G Axes: Nos da la posibilidad de poder guardar las gráficas generadas en el proceso, y con los botones del cuadro de datos generados elegimos que grafica queremos guardar.
 - Graficar: Sirve para llevarnos a la última GUI, modulo 3.
 - Datos Excel: Utilizado para abrir el Excel base2.xlsx donde se guardan todo los datos generados considerados de interés.
 - Valores de entrada: Botón para abrir el Excel libro0.xlsx donde viene todos los valores de entrada y donde se puede visualizar qué valor se está utilizando en ese proceso concretamente, sin tener que volver a inicio y modificarlo a mano de nuevo.

- Texto editable:
 - Distancia para medir: En un principio tuvo otra utilidad, pero en este momento se utiliza para seleccionar el tramo que se va a representar en la gráfica de arriba en medio.

- Texto estático:
 - Estado de error: Sirve para indicar el estado en que se encuentra el proceso. Al iniciar esta GUI indicara pausa, en el cálculo de Matlab indicara el valor del error, y una vez terminado indicara finalizado. Muestra el estado en que se encuentra el programa en todo momento.
 - Tamaño total de la tubería: Viene indicado la longitud total de la tubería que se está analizando.

- Botón de selección: (Toggle Button) Tenemos dos grandes recuadros de selección:
 - Selección: Será el botón que seleccionara que parámetros se van a graficar en las 4 graficas que tenemos en pantalla. Los parámetros seleccionables son:
 - Q_c , transferencia de calor que sufrirá el fluido.
 - Q , es el calor suministrado, será de valor constante ya que viene puesto como una constante de entrada.
 - P , la presión, que tendrá una gran importancia en los resultados obtenidos, ya que sabemos la importancia de la caída de presión en las tuberías.
 - m , flujo másico, el cual sabremos su valor en la condición de entrada seleccionada.
 - T_w , temperatura de la pared.
 - T , temperatura del fluido.
 - H , entropía del fluido en el tubo.
 - h , entalpia.
 - x_0 , calidad del fluido, indicado su estado en liquido o gas según su valor.
 - VF , por último tenemos el valor de fracción de vacío, el cual es de gran importancia y sobre todo para saber en qué zona se empieza a dar y cuál es su valor en cada instante.

- Datos generados. En este panel podremos seleccionar que grafica de las cuatro que tenemos queremos guardar. Se bloqueara cuando el proceso esté en ejecución pero volverá a estar disponible cuando se finalice.

6.4. Graficar (módulo 3).

Parte final del programa **Casal** o su último modulo, fue un complemento añadido al final de desarrollo del programa.

Esta se GUI se utilizara exclusivamente para poder observar de una forma visual los valores y ver los puntos más determinantes sin tener que utilizar otro programa externo para ello. De gran utilidad ya que vemos las gráficas generadas por el programa para cualquier parámetro y con la condición de momento o emplazamiento que queramos.

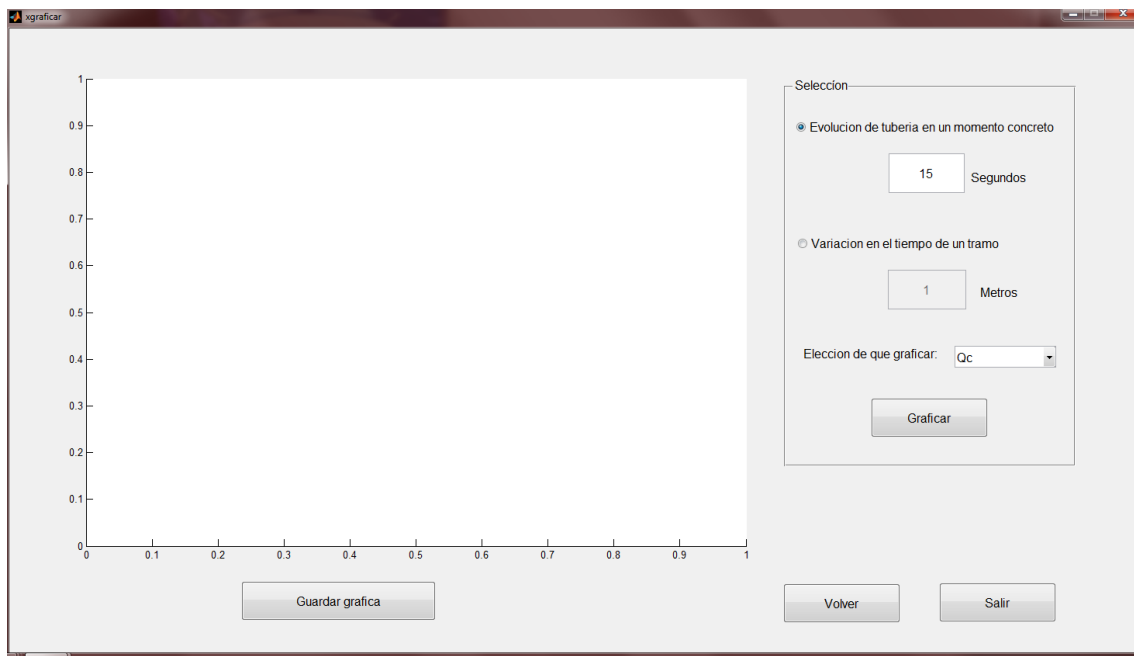


Figura 16 GUI xzgraficar.

Consiste realmente en representar los valores que se tienen guardado de salida en la tabla de Excel base2.xlsx y reproducirlo en gráficas. Su misión y utilidad radica en que esto se puede realizar sin la necesidad de volver a calcular en la GUI anterior y por tanto ahorrando mucho.

Los objetos más importantes son los siguientes:

- Push buttons:
 - Volver: Se vuelve a la GUI xzoperaciones.
 - Salir.
 - Guardar graficar: guardar la gráfica generada en esta GUI con el nombre que se quiera y con el formato que se desee.
 - Graficar: transformar los datos del Excel, generado anteriormente en GUI de xzoperaciones, graficando sus valores.

- Texto editable:
 - Segundos: indicamos el momento en que queremos graficar la tubería completa.
 - Metros: indicamos el tramo de tubería en que queremos ver su evolución en el tiempo.

- Pop-up Menu:
 - Elección de que graficar: Como en la GUI anterior podremos elegir que parámetros mirar. Q_c , Q , P , m , Tw , T , h , H , x_0 , VF .

- Botón de selección: (Toggle Button):
 - Sera una elección sencilla, o elegimos un tramo en el que queremos estudiar su evolución en el tiempo o elegimos un momento en el que queremos ver la distribución en la tubería entera.

6.5. Excel.

Para la realización del entorno visual que se analiza en este documento, se ha empleado como soporte de ayuda, tablas de Excel exteriores para el guardado de los valores de entrada y para valores de salida. Para un mejor manejo de ellos y poder disponer de todo los valores calculados, a disposición del usuario, una vez finalizado la simulación.

Por tanto tendremos dos Excel implementados en el programa, libro1.xlsx que serán los valores de entrada y base2.xlsx que son los resultados generados en el programa.

6.5.1. Excel de valores de entrada.

Se puede abrir desde la GUI xzoperaciones como se ha indicado anteriormente desde el boton valores de entrada.

Un ejemplo practico seria el siguiente, donde podemos ver que en la columna A es un pequeño resumen de que valor se introduce o hacen referencia ,para dar luego su valor utilizado en el programa en la columna B.Solo tiene una pestaña donde vendra toda la informacion y valores utilizados por **Casal**.

La tabla de excel utilizada para este uso se llama libro1.xlsx.

Longitud(m)	1,5
Ancho(m)	0,012
Inclinacion(angulo)	-90
Elementos placa	50
Paso de tiempo(seg)	0,5
N pasos de tiempo maximos	100
Tiempo maximo(seg)	50
Tolerancia estado estable	0,001
Factor de relajacion para la presion	0,5
Factor de relajacion para la velocidad	1
Buher(EB=1,ES3=2,ES33a=3,ES33b=4,ES54=5)	1
FLUJO	0,109252
TINICIAL(C)	100
TENTRADA(C)	100
Presion(pa)	1080000
CALIDAD inicial	0
CALIDAD entrada	0
TIPO DE FLUIDO	13
*(NADA)	0
Q flujo de calor(Kw/m^2)	300
Tener en cuenta pared si o no (T o F)	1
Conductividad Termica de la placa (W/Mk)	372
Calor especifico de la placa (J/kgk)	389
Densidad de la placa (kg/m^3)	8900
Espesor de placa (m)	0,01
Rutina para el calculo del VOID FRACTION	1
Nombre de rutina para el calculo perdidas	1
Lugar donde se impone condicion de borde	1
relacion del tiempo(complejo 1, simple 2)	1

Figura 17 Excel de entrada de valores.Libro1.xlsx.

6.5.2. Excel de valores de salida.

Una tabla generada una vez que el programa ha finalizado la simulación en la GUI xzoperaciones.

Su diseño es simple, en el cual se crean pestañas con el nombre de cada parámetro calculado. Luego en la columna A vendrá una disposición del emplazamiento en que se ha analizado la tubería, indicando su distancia real en metros .En la fila 1 se marca el número de iteración en la cual se ha generado el valor obtenido.

Un ejemplo de esta tabla seria la siguiente utilizando la tabla de valores de entrada anteriormente utilizada. La tabla tiene el nombre de base2.xlsx

	A	B	C	D	E	F	G	H	I	J
1	zona(m) / tiempo (sg)	1	2	3	4	5	6	7	8	9
2	0	21,5790097	54,5110363	96,6624275	147,59535	206,589582	272,276842	360,515256	468,316011	585,460988
3	0,03	21,5790097	54,5110363	96,6624275	147,59535	206,589582	272,276842	360,515256	468,316011	585,460988
4	0,06	20,9312022	52,4506706	92,3735387	140,14858	194,98085	255,543799	335,290137	430,865312	533,15027
5	0,09	20,3073334	50,4744358	88,278909	133,0711	183,995139	239,770214	311,708196	396,199645	485,176048
6	0,12	19,706528	48,5790339	84,3705452	126,347454	173,606603	224,917716	289,697213	364,183948	441,313803
7	0,15	19,1279416	46,7612791	80,6406678	119,962363	163,789022	210,94585	269,179015	334,667527	401,303541
8	0,18	18,5707602	45,0180962	77,0817245	113,900812	154,516178	197,813166	250,07278	307,493312	364,871698
9	0,21	18,0341987	43,3465186	73,6863991	108,148138	145,762134	185,47802	232,297254	282,503723	331,744105
10	0,24	17,5175002	41,7436864	70,447618	102,690072	137,501445	173,899148	215,772238	259,544249	301,653769
11	0,27	17,019935	40,2068437	67,3585498	97,5127817	129,709307	163,036076	200,419659	238,465861	274,345477
12	0,3	16,5407996	38,7333363	64,4126108	92,6028951	122,361663	152,849419	186,164115	219,126382	249,578385
13	0,33	16,0794161	37,320609	61,6034608	87,9475188	115,435279	143,301073	172,93343	201,39124	227,127335
14	0,36	15,6351309	35,9662027	58,9250009	83,5342461	108,907793	134,354403	160,65884	185,133984	206,783291
15	0,39	15,207314	34,6677518	56,37137	79,3511604	102,757743	125,974236	149,275149	170,236263	188,353266
16	0,42	14,7953586	33,4229815	53,9369403	75,3868333	96,9645808	118,127004	138,72076	156,587807	171,659849
17	0,45	14,3986795	32,2297045	51,616312	71,6303202	91,508676	110,780721	128,937628	144,086204	156,540518
18	0,48	14,0167132	31,0858187	49,4043073	68,0711516	86,3712811	103,904969	119,87128	132,636585	142,84682
19	0,51	13,6489164	29,9893039	47,2959643	64,6993235	81,5346135	97,4709215	111,470596	122,151274	130,443467
20	0,54	13,2947659	28,9382195	45,2865309	61,5052857	76,9816753	91,4512694	103,687761	112,549399	119,207419
21	0,57	12,9537574	27,9307014	43,3714576	58,4799282	72,6963736	85,8201973	96,4780963	103,756481	109,026953
22	0,6	12,6254051	26,9649596	41,5463911	55,6145682	68,6634271	80,5533336	89,7999206	95,7040265	99,8007627
23	0,63	12,3092411	26,0392752	39,8071675	52,9009351	64,8683503	75,6276967	83,6143907	88,3291206	91,4370924
24	0,66	12,0048145	25,1519983	38,1498054	50,3311557	61,2974236	71,0216385	77,8853486	81,5740325	83,8529166
25	0,69	11,7116909	24,301545	36,5704992	47,8977395	57,9376628	66,7147635	72,5791684	75,385837	76,9731702
26	0,72	11,4294521	23,4863953	35,0656126	45,5935722	54,7767872	62,6879542	67,6645977	69,716045	70,7300228
27	0,75	11,1576949	22,7050907	33,6316717	43,411859	51,8031918	58,9231966	63,1126222	64,5202713	65,0622248
28	0,78	10,8960311	21,9562312	32,265359	41,3461834	49,0059114	55,4035904	58,8963133	59,7579011	59,9144824
29	0,81	10,644087	21,2384743	30,9635068	39,3904358	46,3745938	52,1132781	54,9906967	55,3917913	55,2368991
30	0,84	10,4015021	20,5505318	29,7230914	37,538831	43,8994685	49,0373872	51,3726203	51,3879859	50,9844571
31	0,87	10,1679298	19,8911683	28,5412267	35,785837	41,5713213	46,1619786	48,0206342	47,7154565	47,1165514
32	0,9	9,94303585	19,2591985	27,4151589	34,1262404	39,3814609	43,4739904	44,9148688	44,3458527	43,5965545
33	0,93	9,72649847	18,6534861	26,3422606	32,5550857	37,3216954	40,9611896	42,0369271	41,2532798	40,3914313
34	0,96	9,51800774	18,0729409	25,3200255	31,0676735	35,3843048	38,6121236	39,3697808	38,4140892	37,471385
35	0,99	9,31726517	17,5165179	24,3460631	29,6595478	33,5620169	36,4160737	36,8976718	35,8066864	34,8095374
36	1,02	9,12398329	16,983215	23,4180938	28,3264847	31,8479819	34,3630111	34,6060213	33,4113545	32,3816402
37	1,05	8,93788526	16,4720712	22,5339439	27,064481	30,2357508	32,4435553	32,4813438	31,2100914	30,165814

Figura 18 Excel de resultados. base2.xlsx

7. Resultados.

En este capítulo se exponen los resultados obtenidos para algunos casos de aplicación. Los siguientes apartados corresponden a ensayos realizados por el programa **Casal**, de los cuales se sabían los valores de salida que deberían dar, de forma aproximada, y así podemos colaborar su buen funcionamiento y su utilidad.

En todos los casos siempre pondremos un número de iteraciones superiores a las requeridas, de manera de garantizar que se alcance el estado estacionario, y por tanto ya totalmente desarrollados.

7.1. Comparación de tiempo de cómputo con diferentes modelos de Runge-Kutta.

En este ejemplo se realizaron pruebas del modelo, variando el orden de precisión del esquema de Runge-Kutta, determinando el tiempo de cómputo total necesario para alcanzar el estado estacionario.

Para la evaluación de programa **Casal** se reproducirá el experimento de Bartolomei (10), el cual consiste en un flujo de agua en condiciones de subenfriamiento en una tubería con orientación -90 grados (vertical de flujo ascendente), el material considerado para el tubo es acero, el cual tiene de conductividad térmica 372 W/Mk, calor específico 389 J/KgK y una densidad de 8900 kg/m³. La tubería mide 1,5 metros, tiene un diámetro de 0.012 metros y un espesor de pared de 0,01 metros. El dominio se dividió en 50 elementos y se realizaron un total de 50 iteraciones temporales con un paso de tiempo de 0.5 s. Se introducirá una temperatura inicial de 263 C, una presión de 10.8 MPa y un flujo de entrada de 0.109252kg/s.

La simulación se realizara para tres flujos de calor en la pared exterior del tubo distintas (300,400 y 450Kw/m²), variando el esquema de integración temporal disponibles en el programa, EB, ES3, ESSa, ES33b y ES54.

Los valores de entrada serán los de la siguiente tabla:

Longitud (m).	1,5
Ancho (m).	0,012
Inclinación (ángulo).	-90
Elementos placa.	50
Paso de tiempo (seg).	0,5
N pasos de tiempo máximo.	100
Tiempo máximo (seg).	50
Tolerancia estado estable.	0,001
Factor de relajación para la presión.	0,5
Factor de relajación para la velocidad.	1
Butcher (EB=1, ES3=2, ES33a=3, ES33b=4, ES54=5).	1,2,3,4,5
Flujo.	0,109252
T inicial(C).	263
T entrada(C).	263
Presión (pa).	10800000
Calidad inicial.	0
Calidad entrada.	0
Tipo de fluido.	1
Q flujo de calor (Kw/m²).	300
Tener encuentra pared sí o no (T=1 o F=2).	1
Conductividad Térmica de la placa (W/Mk).	372
Calor especifico de la placa (J/kgk).	389
Densidad de la placa (kg/m³).	8900
Espesor de placa (m).	0,01
Rutina para el cálculo del VOID FRACTION.	1
Nombre de rutina para el cálculo pérdidas por fricción.	1
Lugar donde se impone condición de borde (ini2, out1).	1
Relación del tiempo(complejo 1, simple 2)	1

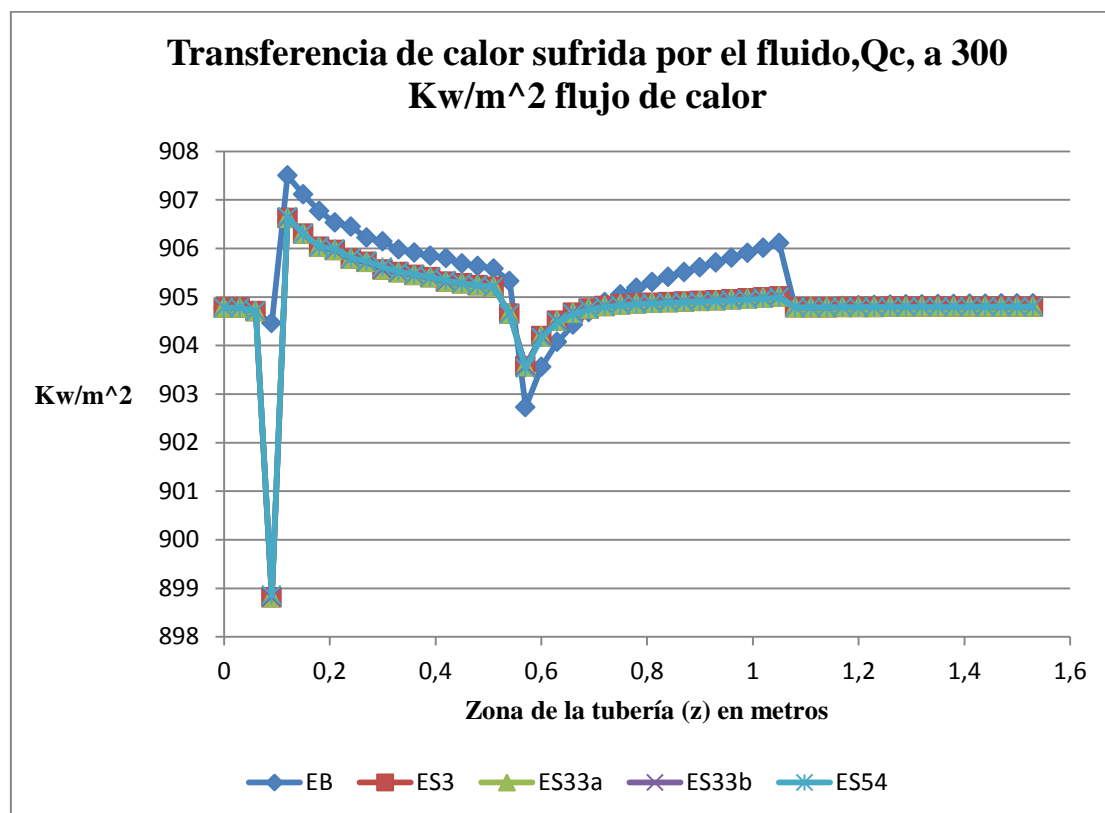
Tabla 1 Valores de entrada primera prueba.

Esta tabla son los valores que introduciremos en el primer caso y solo variaremos el valor de Butcher de 1 a 5.

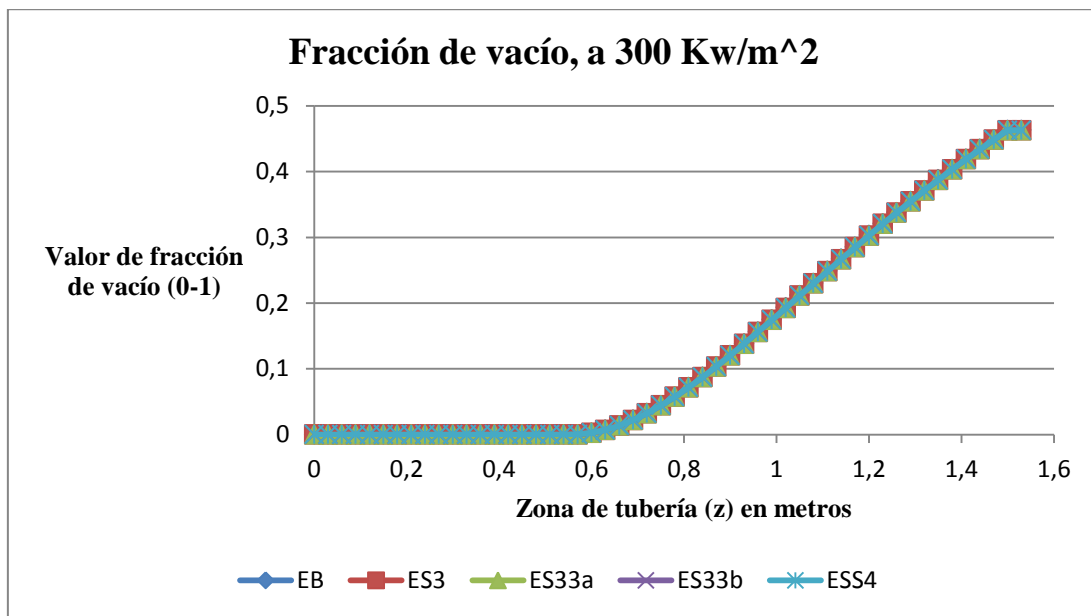
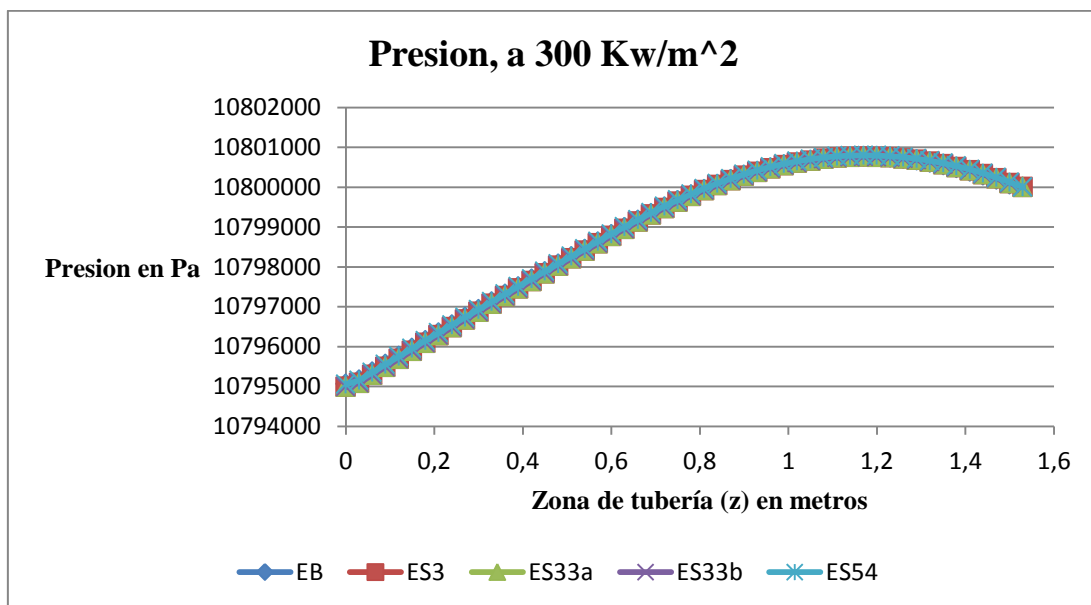
Los tiempos para realizar cada ensayo en el programa **Casal** (tiempo que tarda desde que se inicia la simulación hasta que se alcanza el estado estacionario) son:

- EB (Euler Backward): 91.44 segundos.
- ES3 (ESDIRK-3 Bill y Carpenter 2002): 226.84 segundos.
- ES33a (ES33a Skvortsov 2003): 224. 49 segundos.
- ES33b (ES33b R Alexander 2003): 198.25 segundos.
- ES54 (ES54 Slvortsov 2003): 348.20 segundos.

Se adjunta los diagramas con los valores del calor total transferido al fluido (Q_c), fracción de vacío y presión a lo largo de la tubería cuando se varía el esquema de integración temporal.



Gráfica 1 Transferencia de calor sufrida por el fluido, a 300 Kw/m^2 .

Gráfica 2 Fracción de vacío a 300 Kw/m².Gráfica 3 Presión a 300Kw/m²

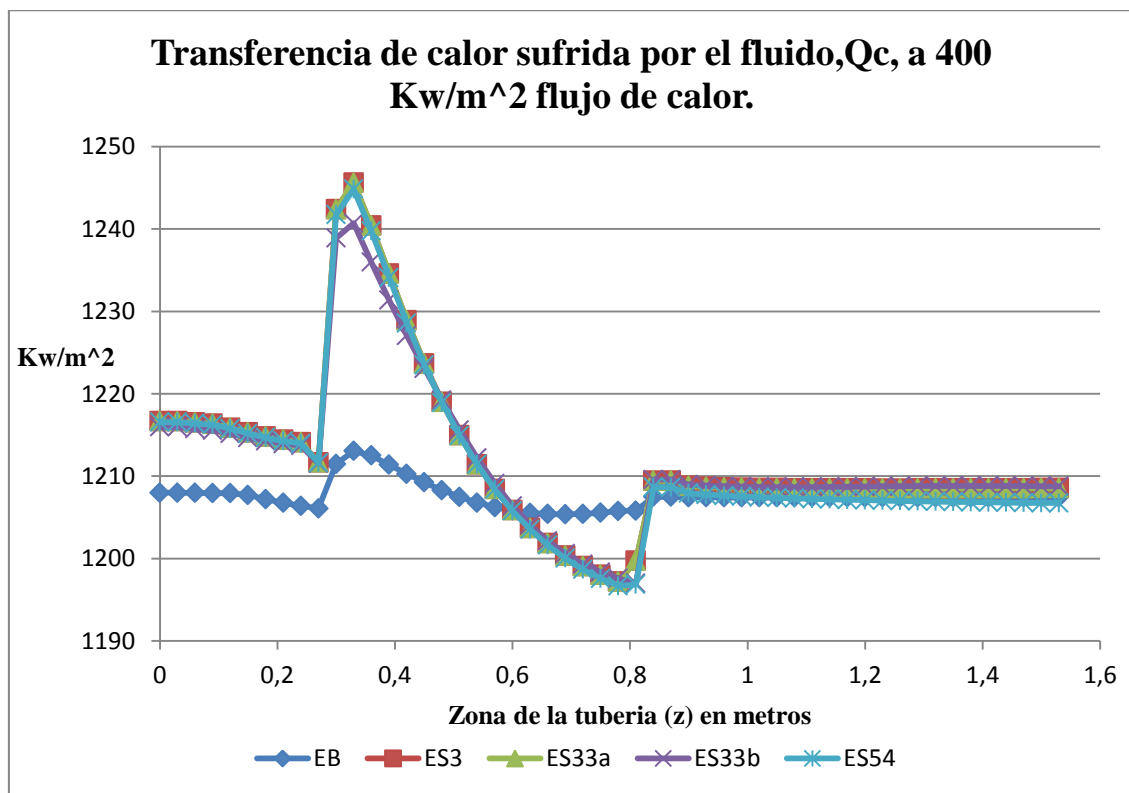
En las gráficas 1 a la 3 se observa en el primer caso una pequeña variación del calor total transferido al fluido cuando se varía el orden de precisión. Esta variación es más pronunciada en el caso del esquema EB. Esta variación es debida a que este esquema al ser de primer orden no captura totalmente la física, necesitando probablemente una

disminución del paso temporal, capturando así más información., en los demás casos no se observa una desviación significativa.

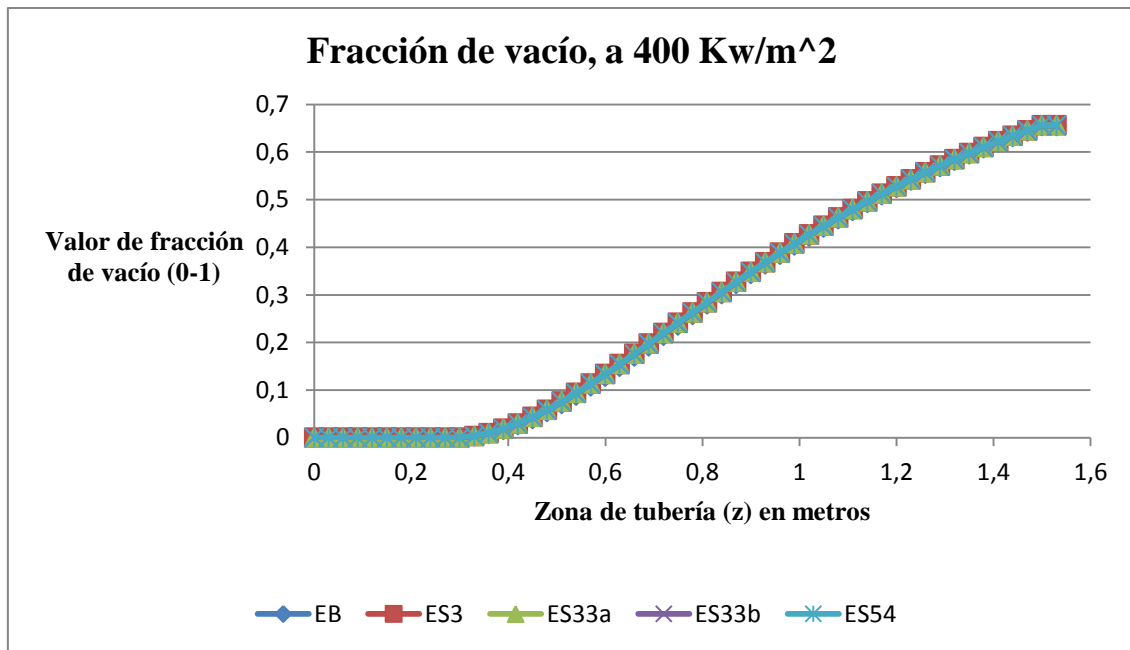
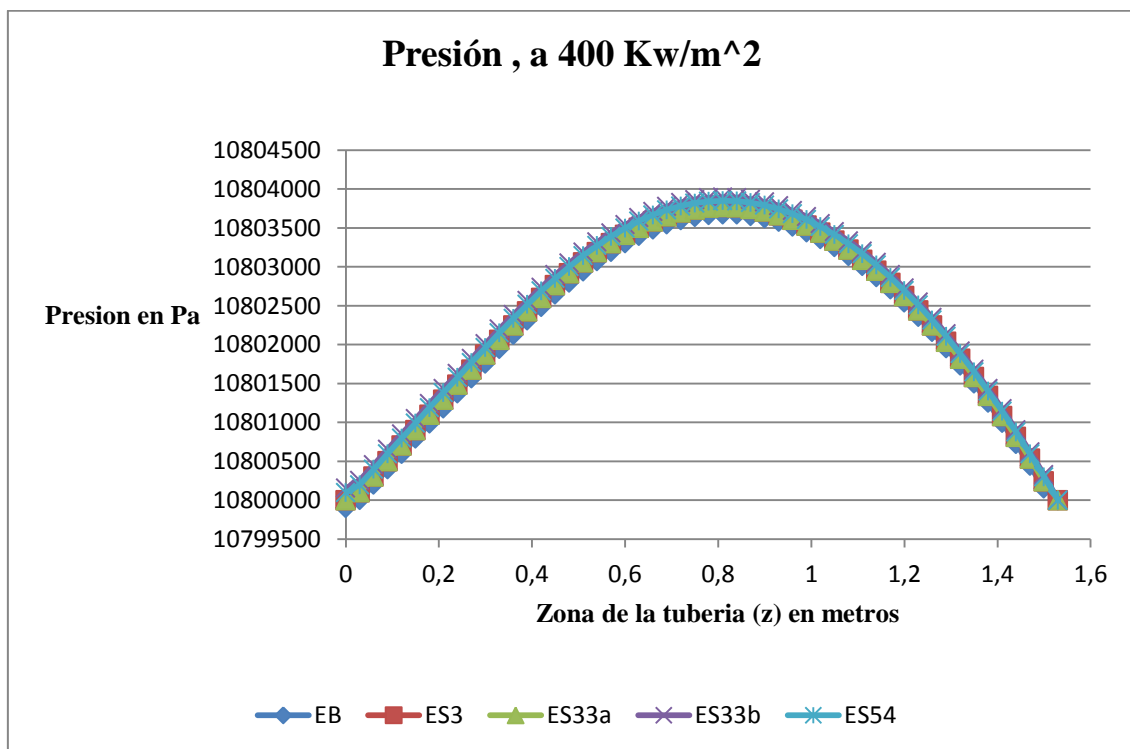
En las dos siguientes gráficas no se observa diferencia alguna en los perfiles de presión y fracción de vacío una vez se alcanzó el estado estacionario, esto se debe que una vez se haya alcanzado dicho estado el efecto transitorio se desvanece y ya el esquema temporal no juega ningún papel, igualmente se infiere de acuerdo a esta semejanza en la solución que los errores temporales no se propagaron en el tiempo.

En este caso repetiremos la misma simulación con los mismos valores, solamente que cambiando el flujo de calor por 400KW/m^2 .

- EB: (EULER BACKWARD) : 138.204566 segundos
- ES3: (ESDIRK-3 BILL & CARPENTER 2002) : 409.000003 segundos
- ES33a: (ES33a SKVORTSOV 2003):420.86767 segundos.
- ES33b: (ES33b R. Alexander 2003): 397.452845segundos
- ES54: (ES54 SKVORTSOV 2003):712.366970 segundos



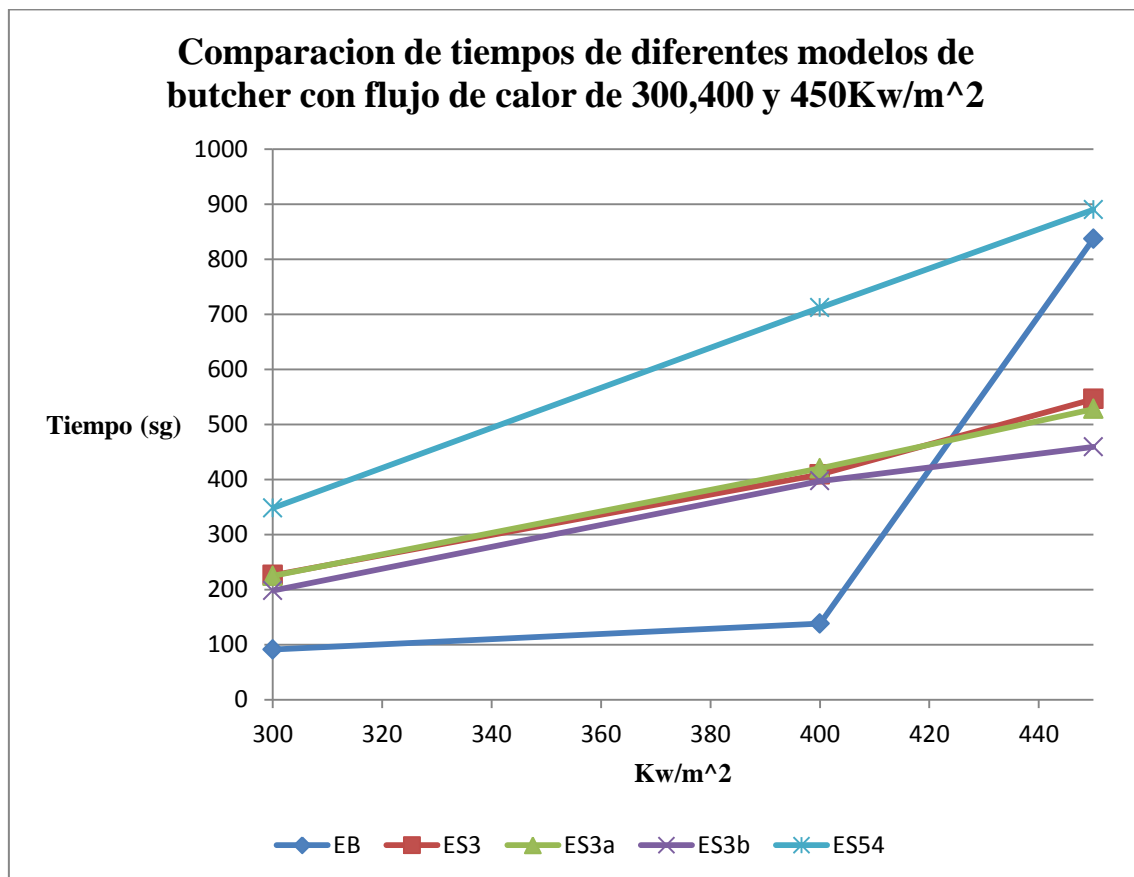
Gráfica 4 Transferencia de calor sufrida por el fluido, a 400 Kw/m^2 .

Gráfica 5 Fracción de vacío a 400 Kw/m².Gráfica 6 Presión a 400Kw/m².

En las gráficas anteriores igualmente se observa la poca dependencia en las soluciones estacionarias, sin embargo el valor del calor transferido al fluido si varió más dramáticamente, lo que implica que orden de precisión temporal incide en la solución transitoria conforme las condiciones de flujo de calor en la pared sean más extremas.

Repitiendo el mismo ensayo pero con el flujo de calor en 450Kw/m^2 se obtienen parecidas gráficas, solo cambia los tiempos.

Se observa que se obtiene valores parecidos de resultados utilizando un modelo u otro de Runge-Kutta, que solo se tiene valores diferentes para Euler, que en los cambios de fases no sufre tanta variación como para los otros modelos.



Gráfica 7 Comparación de tiempos de un modelo a otro.

Podemos ver que además el tiempo de cómputo para el modelo EB, en un principio, siempre es menor hasta que se empieza a introducir altos valores de flujo de calor en el cual otros modelos como sería el caso de ES3b tardan menos y dan similares valores. Demostrando que cada modelo de Runge-Kutta es interesante la utilización de uno u otro dependiendo de que valores de entrada que vayamos a introducir.

7.2. Diferentes ángulos de inclinación de la tubería.

Se comprueba con el programa **Casal** la variación de la presión dentro de la tubería, en el momento final de la iteración del programa, cuando está totalmente desarrollado, obteniendo sus valores finales.

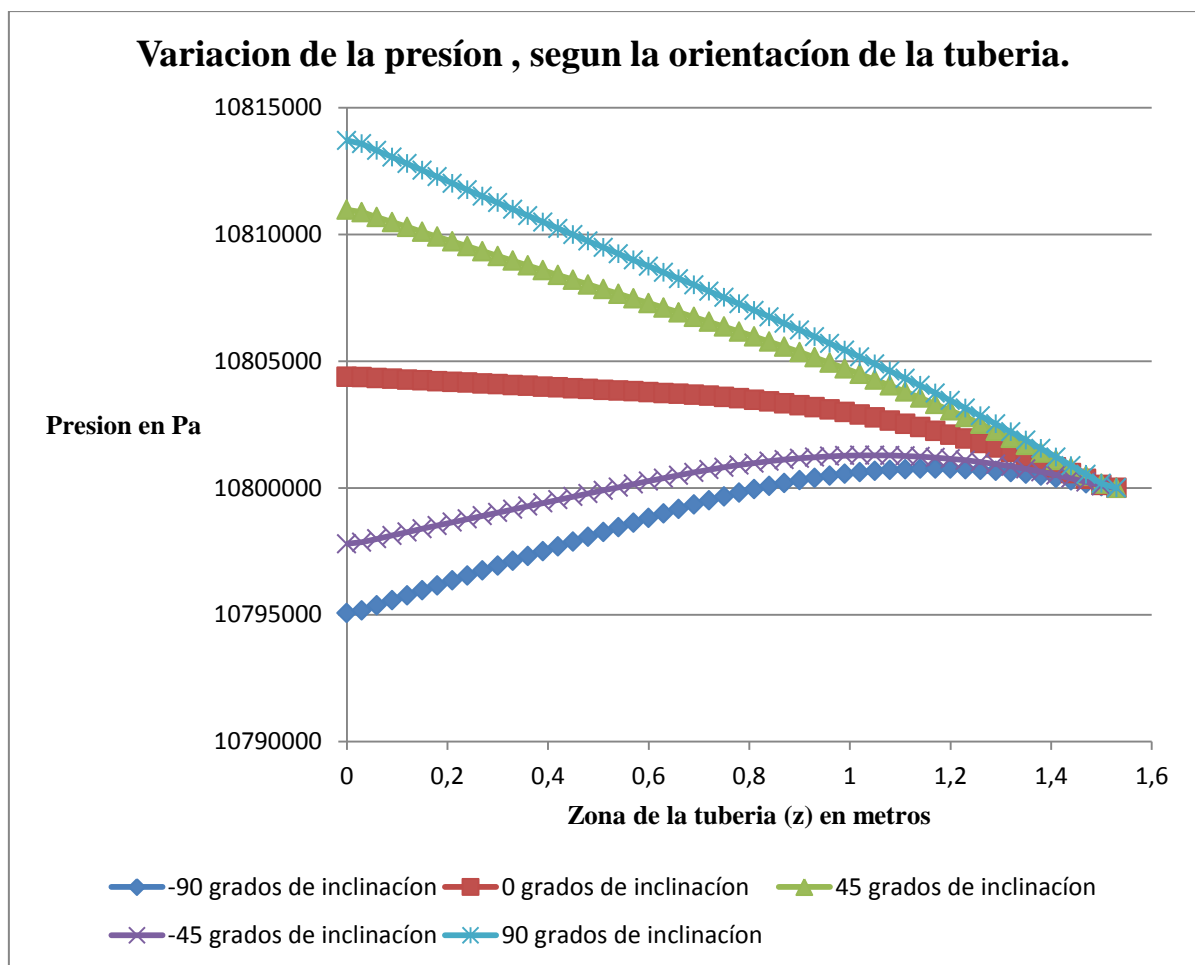
Para la evaluación de programa Casal se reproducirá el experimento el cual consiste en una tubería de material acero, el cual tiene de conductividad térmica 372 W/Mk, calor específico 389 J/KgK y una densidad de 8900 kg/m³. La tubería mide 1,5 metros, tiene un diámetro de 0.012 metros y un espesor de pared de 0,01 metros. El dominio se dividido en 50 elementos y se realizaran un total de 50 iteraciones temporales con un paso de tiempo de 0.5 s. Se introducirá una temperatura inicial de 263 C, una presión de 10.8 MPa y un flujo de entrada de 0.109252kg/s y el flujo de calor suministrado será de 300Kw/m². El fluido utilizado será agua, y por tanto su selección será el valor 1 en las tablas.

Longitud (m).	1,5
Ancho (m).	0,012
Inclinación (ángulo).	-90,0,45,- 45 90
Elementos placa.	50
Paso de tiempo (seg).	0,5
N pasos de tiempo máximo.	100
Tiempo máximo (seg).	50
Tolerancia estado estable.	0,001
Factor de relajación para la presión.	0,5
Factor de relajación para la velocidad.	1
Butcher (EB=1, ES3=2, ES33a=3, ES33b=4, ES54=5).	1
Flujo.	0,109252
T inicial(C).	263
T entrada(C).	263
Presión (pa).	10800000
Calidad inicial.	0
Calidad entrada.	0
Tipo de fluido.	1
Q flujo de calor (Kw/m ²).	300

Tener encuentra pared sí o no (T=1 o F=2).	1
Conductividad Térmica de la placa (W/Mk).	372
Calor específico de la placa (J/kgk).	389
Densidad de la placa (kg/m^3).	8900
Espesor de placa (m).	0,01
Rutina para el cálculo del VOID FRACTION.	1
Nombre de rutina para el cálculo pérdidas por fricción.	1
Lugar donde se impone condición de borde (ini2, out1).	1
Relación del tiempo(complejo 1, simple 2)	1

Tabla 2 Para la segunda prueba.

Para ello se introducirá los mismos valores de entrada que en la tabla 2 pero se modificara para cada prueba el valor de inclinación de -90, -45, 0, 45 y 90 grados.



Gráfica 8 Comparación de variación de presión según ángulo de inclinación de tubería.

Como se puede observar claramente, las gráficas de variación de presión dentro de la tubería cambian según la inclinación, esto es debido a que según esta posición tendrá a favor o en contra la fuerza de la gravedad y por tanto dará unos valores u otros en su caída de presión. En los casos de inclinación con ángulo negativo aumenta la presión ya que a ser empujada por la gravedad hace que aumente, mientras que en caso sin inclinación o con esta en valor positivo se puede observar claramente la caída típica de la presión en fluido.

7.3. Diferentes flujos de calor suministrados.

En estos casos, se mirara la evolución del valor de la presión y de la fracción de vacío a diferentes valores de flujo de calor.

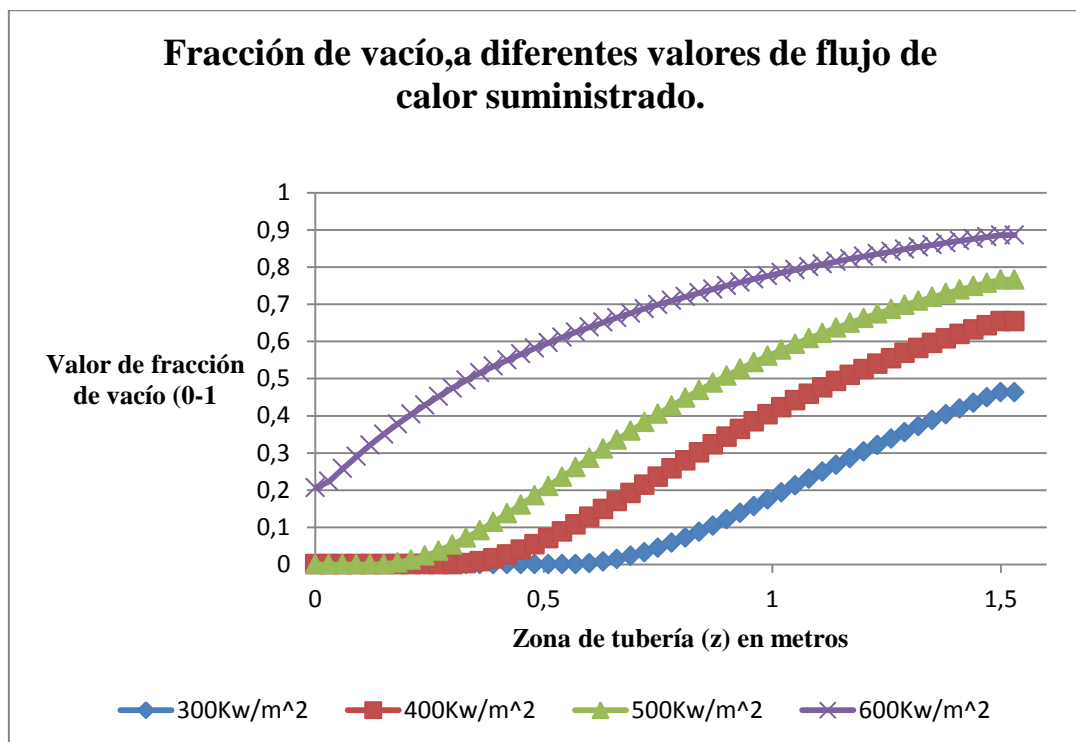
Los datos de entrada serán de una tubería con orientación -90 grados, el material considerado para el tubo es acero, el cual tiene de conductividad térmica 372 W/Mk, calor específico 389 J/KgK y una densidad de 8900 kg/m³. La tubería mide 1,5 metros, tiene un diámetro de 0.012 metros y un espesor de pared de 0,01 metros. El dominio se dividió en 50 elementos y se realizaron un total de 50 iteraciones temporales con un paso de tiempo de 0.5 s. Se introducirá una temperatura inicial de 263 C, una presión de 10.8 MPa y un flujo de entrada de 0.109252kg/s y el flujo de calor suministrado será de 300, 400, 500 y 600 Kw/m². El fluido utilizado será agua, y por tanto su selección será el valor 1 en las tablas.

Longitud (m).	1,5
Ancho (m).	0,012
Inclinación (ángulo).	-90
Elementos placa.	50
Paso de tiempo (seg).	0,5
N pasos de tiempo máximo.	100
Tiempo máximo (seg).	50
Tolerancia estado estable.	0,001
Factor de relajación para la presión.	0,5
Factor de relajación para la velocidad.	1
Butcher (EB=1,ES3=2,ES33a=3,ES33b=4,ES54=5).	1

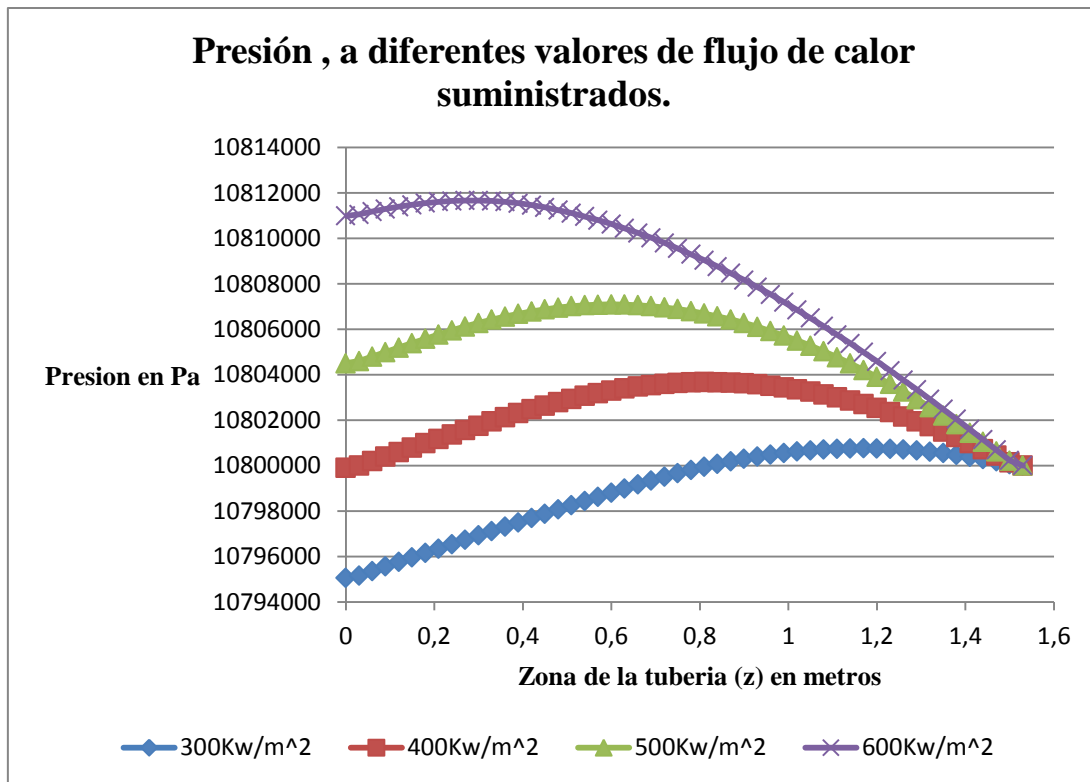
Flujo.	0,109252
T inicial(C).	263
T entrada(C).	263
Presión (Pa).	10800000
Calidad inicial.	0
Calidad entrada.	0
Tipo de fluido.	1
Q flujo de calor (Kw/m ²).	300,400,500,600
Tener encuentra pared si o no (T=1 o F=2).	1
Conductividad Térmica de la placa (W/Mk).	372
Calor especifico de la placa (J/kgk).	389
Densidad de la placa (kg/m ³).	8900
Espesor de placa (m).	0,01
Rutina para el cálculo del VOID FRACTION.	1
Nombre de rutina para el cálculo pérdidas por fricción.	1
Lugar donde se impone condición de borde (ini2, out1).	1
Relación del tiempo(complejo 1, simple 2)	1

Tabla 3 Prueba con diferentes flujos de calor.

En esta ocasión lo que se va a analizar es la variación de la presión y el valor de la fracción de vacío.



Gráfica 9 Fracción de vacío, a diferentes valores de flujo de calor suministrado.



Gráfica 10 Presión, a diferentes valores de flujo de calor suministrados.

Se observa en la, grafica 9, que al aumentar el flujo de calor en la tubería el valor de fracción de vacío aumenta y comienza antes su subida. Ya que está relacionado con él y estos datos son de gran utilidad en la vida real, ya que los valores de fracción de vacío son importantes para evitar fallos en instalaciones.

En la segunda gráfica se observa claramente que aunque en un principio no teníamos caída de presión, debido a que estaba colocada a -90 grados, según aumentamos el flujo de calor suministrado la caída se va a pronunciando más. Ya que en esos valores no tendría en cuenta tanto la fuerza ejercida por la gravedad porque estaría casi en estado gaseoso o gaseoso completamente e influiría otros factores en la caída de presión.

7.4. Pruebas sin tener en cuenta las paredes de la tubería.

Se comprobará el programa **Casal** cuando seleccionamos que no se quiere tener en cuenta las paredes de la tubería. En ese caso el programa considera un valor fijo de flujo de calor transferido directamente al fluido, en cuyo caso no se calcula el coeficiente convectivo.

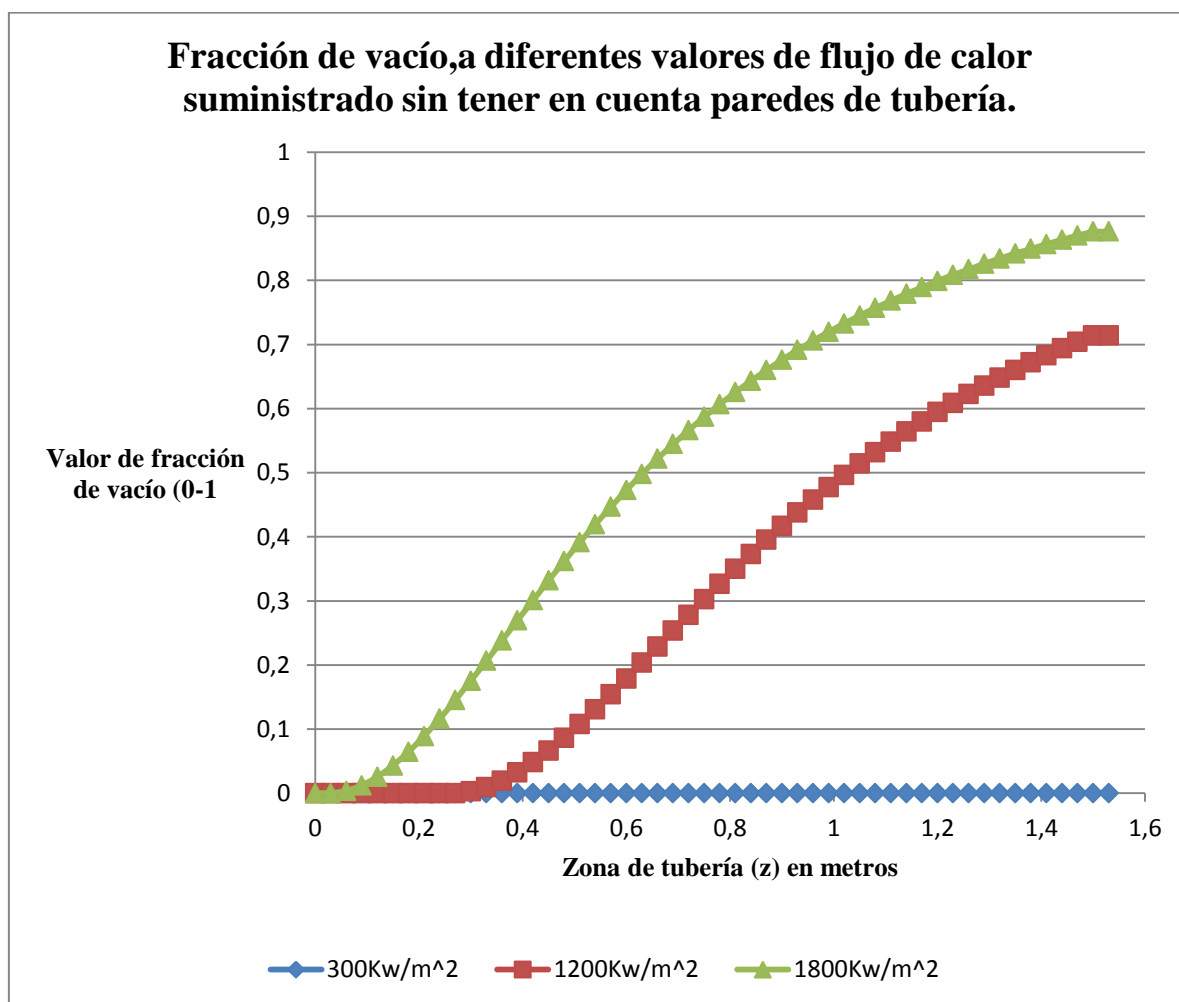
La evaluación de programa Casal consiste en no considerar las paredes de la tubería ni del material que está formado. La pared mide 1,5 metros y una sección de pared de 0,01 metros. El dominio se dividió en 50 elementos y se realizaron un total de 50 iteraciones temporales con un paso de tiempo de 0.5 s. Se introducirá una temperatura inicial de 263 grados, una presión de 10.8 MPa y la entrada del fluido será un flujo de 0,109252 kg/s. El fluido utilizado será agua, y por tanto su selección será el valor 1 en las tablas. Se introducirá valores de flujo de calor de 300,1200 y 1800 Kw/m².

Longitud (m).	1,5
Ancho (m).	0,012
Inclinación (ángulo).	-90
Elementos placa.	50
Paso de tiempo (seg).	0,5
N pasos de tiempo máximo.	100
Tiempo máximo (seg).	50
Tolerancia estado estable.	0,001
Factor de relajación para la presión.	0,5
Factor de relajación para la velocidad.	1
Butcher (EB=1, ES3=2, ES33a=3, ES33b=4, ES54=5).	1
Flujo.	0,109252
T inicial(C).	263
T entrada(C).	263
Presión (pa).	10800000
Calidad inicial.	0
Calidad entrada.	0
Tipo de fluido.	1
Q flujo de calor (Kw/m ²).	300,1200,1800
Tener encuentra pared sí o no (T=1 o F=2).	2

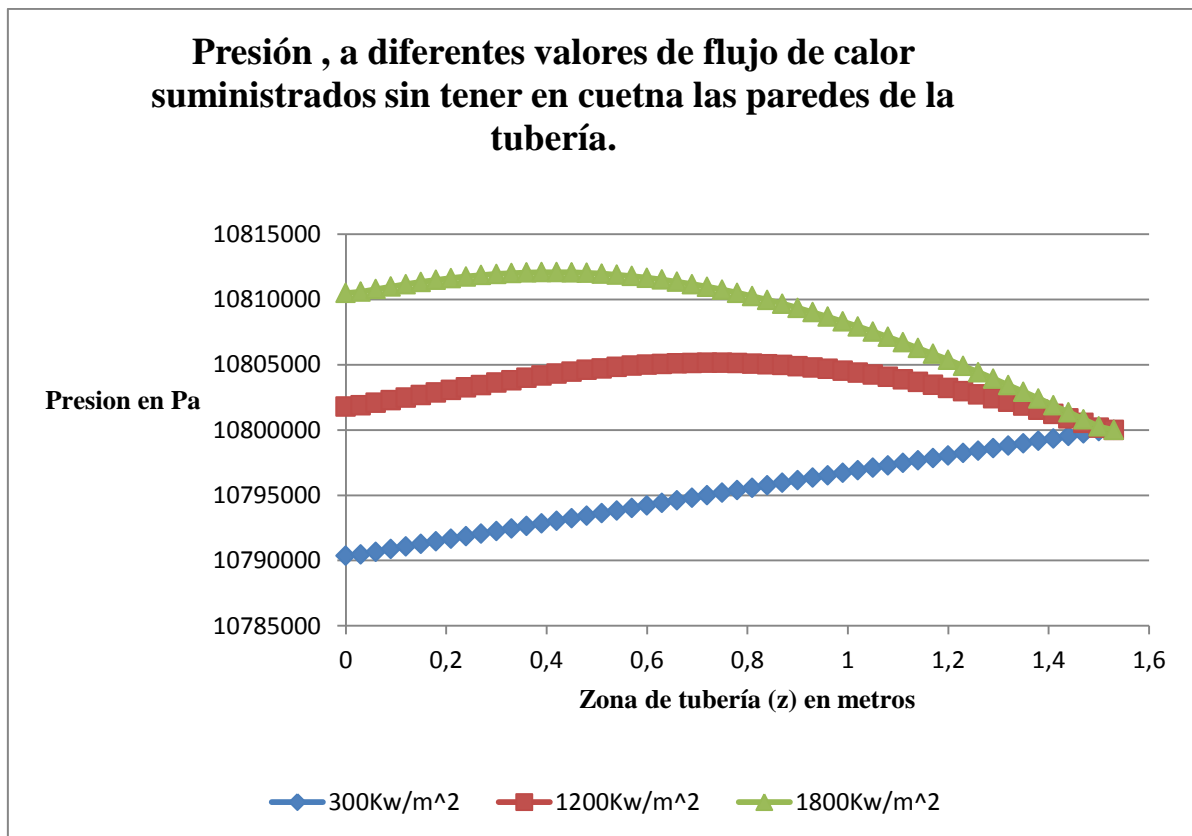
Conductividad Térmica de la placa (W/Mk).	372
Calor específico de la placa (J/kgk).	389
Densidad de la placa (kg/m ³).	8900
Espesor de placa (m).	0,01
Rutina para el cálculo del VOID FRACTION.	1
Nombre de rutina para el cálculo pérdidas por fricción.	1
Lugar donde se impone condición de borde (ini2, out1).	1
Relación del tiempo(complejo 1, simple 2)	1

Tabla 4 Sin tener en cuenta las paredes de la tubería.

Dando como resultado las siguientes gráficas.



Gráfica 11 Fracción de vacío, a diferentes valores de flujo de calor suministrados sin tener en cuenta las paredes de la tubería.



Gráfica 12 Presión, a diferentes valores de flujo de calor suministrado sin tener en cuenta las paredes de la tubería.

Se observa claramente que para obtener resultados, con valores notables comparados con los anteriores casos que si consideramos la pared de la tubería, tenemos que introducir valores significativamente superiores. Llegando al punto de introducir valores como 1800Kw/m² como valor de flujo de calor suministrado, cosas que no se podría realizar en las pruebas en donde si tenemos en cuenta la pared de la tubería, ya que el programa se bloquearía y no realizaría los cálculos.

Los tiempos de cómputo para esta serie de problemas son muy rápido, ya que al no tener en cuenta las paredes de la tubería son cálculos que se ahorra realizar la simulación.

7.5. Diferentes materiales de tubería.

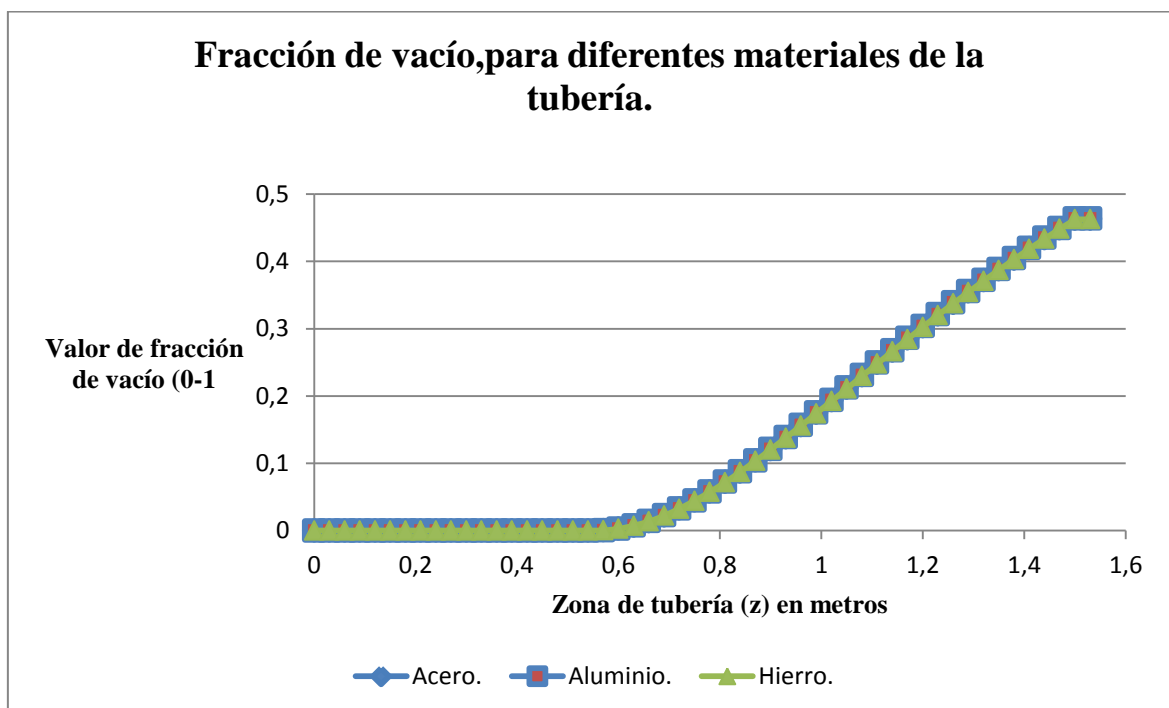
Repetición de experimento anterior, del caso primero pero esta vez variando el material de la tubería

Para esta prueba utilizaremos tres materiales diferentes:

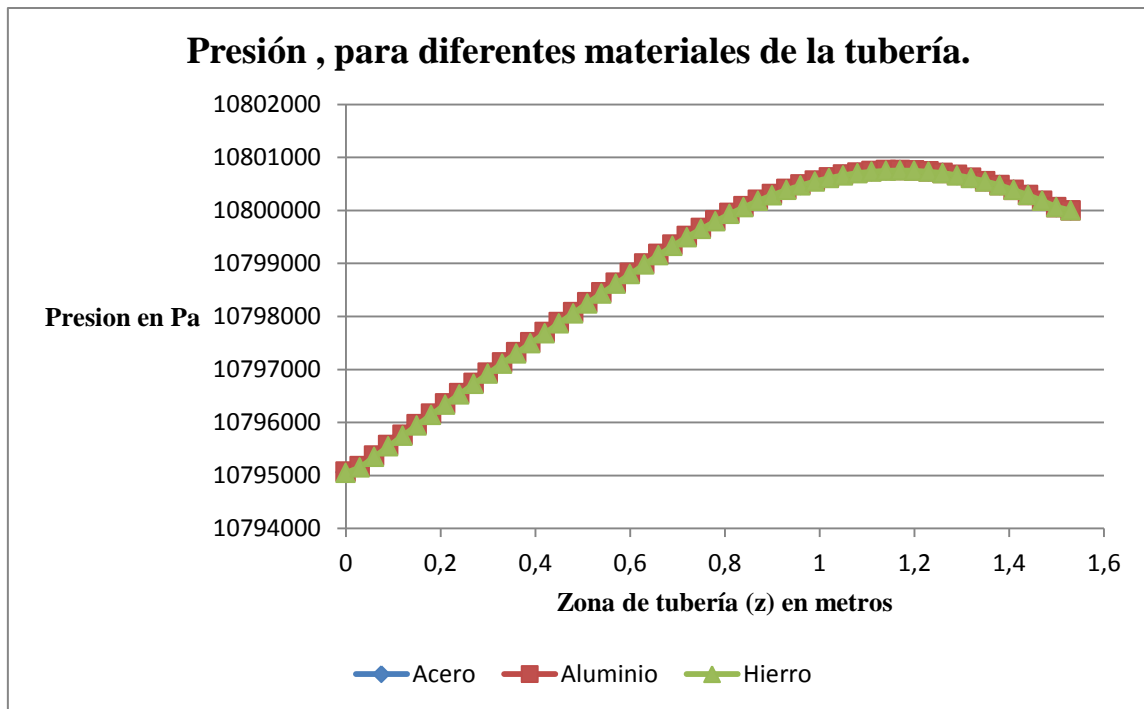
Material:	Conductividad Térmica de la placa (W/Mk).	Calor específico de la placa (J/kgk).	Densidad de la placa (kg/m ³).
Acero.	372	389	8900
Aluminio	72	473	7870
Hierro.	209	909	2700

Tabla 5 Diferentes valores de materiales.

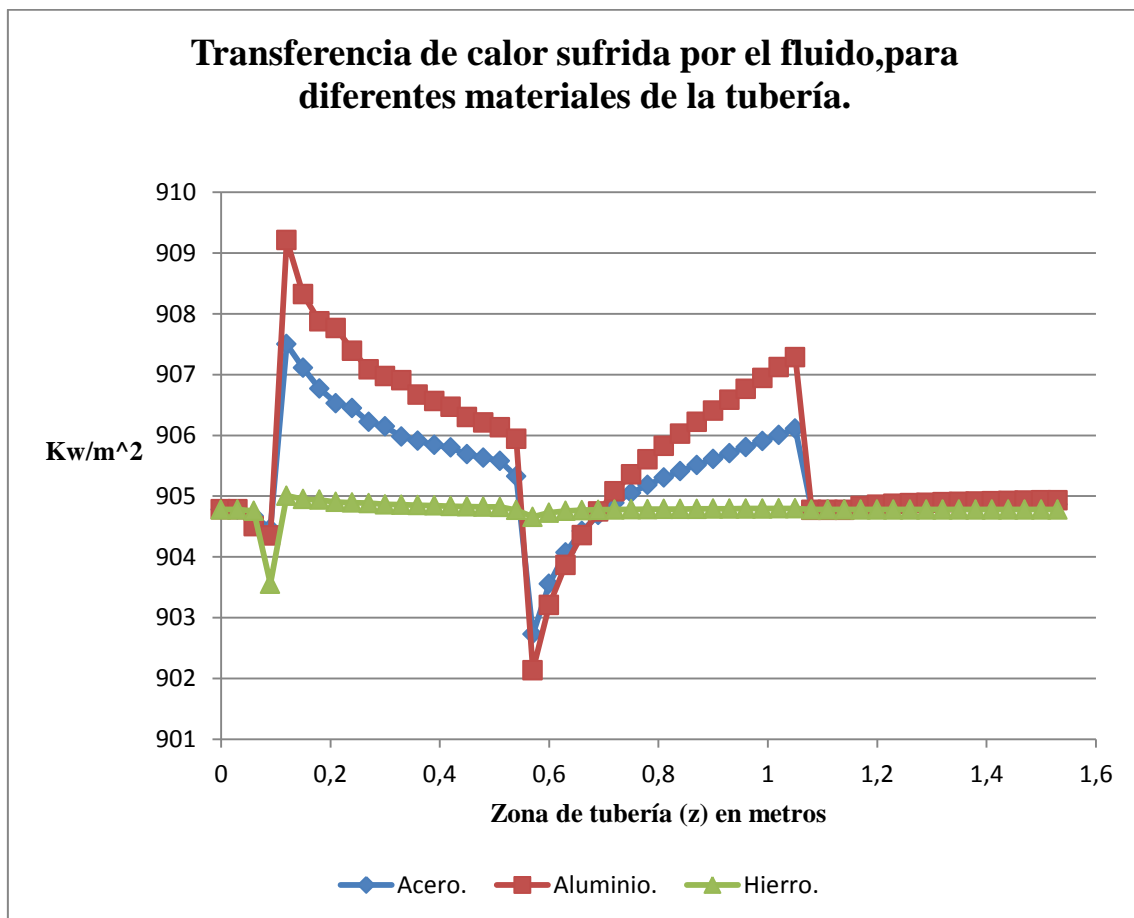
Para la evaluación de programa Casal se reproducirá el experimento el cual consiste en una tubería con orientación -90 grados, de un material diferente en cada ensayo. La tubería mide 1,5 metros, tiene un diámetro de 0.012 metros y un espesor de pared de 0,01 metros. . El dominio se dividió en 50 elementos y se realizaron un total de 50 iteraciones temporales con un paso de tiempo de 0.5 s. Se introducirá una temperatura inicial de 263 C, una presión de 10.8 MPa, un flujo de entrada de 0.109252kg/s y el flujo de calor suministrado será de 300Kw/m².



Gráfica 13 Fracción de vacío para diferentes materiales de tubería.



Gráfica 14 Presión para diferentes materiales de la tubería.



Gráfica 15 Transferencia de calor sufrida por el fluido para diferentes materiales de la tubería.

Como se puede observar por las tres gráficas, la variación del material que se seleccione para la tubería repercutirá sobre todo en los valores obtenidos para transferencia de calor sufrida por el fluido, dicha variación es debida gracias a la variación en la conductividad térmica la cual incide en el cálculo del coeficiente convectivo.

La presión y el valor de fracción de vacío no sufrirán grandes variaciones debido a esta elección ya que la rugosidad no varía mucho de acuerdo al material considerado.

8. Conclusiones.

El objetivo de este proyecto es el de optimizar y desarrollar el entorno visual de un programa computacional, para un programa escrito en Matlab anteriormente. Teniendo en cuenta en su elaboración que debería ser de fácil utilización por los demás usuarios y accesible, pudiendo ser usado por personas sin muchos conocimientos en programación, se puede concluir de manera general que:

- El entorno visual desarrollado elimina la necesidad de que el usuario posea altos conocimientos en programación, pudiendo ejecutar el mismo de manera fácil y sencilla.
- El programa Casal puede servir como una herramienta para el análisis y estudio de evaporadores u otro tipo de intercambiador en donde ocurra cambio de fase.
- Se puede obtener muchísima información física de un problema que puede ser de gran utilidad en problemas de intercambiadores de calor.
- Igualmente, el programa desarrollado podrá ser una herramienta de carácter docente para el estudio de flujos con cambio de fase.

Y en cuanto a los resultados de los problemas estudiados:

- El programa Casal reproduce satisfactoriamente el comportamiento de flujos con cambio de fase liquido-vapor.
- El tiempo de cómputo para el caso considerado depende en gran medida del esquema de integración considerado.

9. Futuras líneas de trabajo y desarrollos futuros.

La optimización y desarrollo del entorno visual de un programa computacional es un proceso largo y muy complejo, siempre lleno de ideas nuevas y cosas que queremos incluir dentro del programa para su mejora, y cosas que mejorar dentro del código base que se verán reflejado en los tiempos de cálculo y sobre todo de resolución de estudio para su mejor uso.

Para ello, se tiene que implicar un trabajo con continuidad en el desarrollo, que partiendo del trabajo aquí expuesto se puede evolucionar y optimizar el conjunto desde concretar más las limitaciones de introducción de valores de entrada, para que siempre se pongan valores en el cual el programa lo desarrolle siempre de forma fluida y óptima, hasta modificación de los datos mostrados en la pantalla del entorno visual.

Para las limitaciones de valores de entrada se deberán hacer muchas más pruebas con cada fluido, probando sus límites superiores e inferiores, y modificando valores de entrada para ver cómo se comporta el programa y optimización en la forma de escritura dentro del código. En resumidas cuentas, dejarla más pulida y haciendo que funcione mejor, con lo que implica limitar el uso de valores no óptimos a través de mensajes de error que no te dejen guardar los datos si no son aceptables.

También añadir más opciones en el entorno visual, como sería la introducción de un reloj interno para marcar el tiempo que tarda en procesar los datos. Para saber cuánto tarda y que éste se reflejara en la pantalla, de una forma que el usuario pueda obtener esta información por si le es de utilidad.

Por tanto, quedaría mucho que hacer, sobre todo en la parte de optimización del entorno visual, utilizando el ya creado como base y mejorándolo.

10. Presupuesto.

Se expone a continuación un pequeño desglose de las actividades, equipos y personal que ha sido utilizada o ha colaborado en la realización de este proyecto, detallándose con información económica y temporal, con objeto de ofrecer un presupuesto que pueda servir como indicador del coste del proyecto.

El desglose de costes se ha dividido en cuatro grupos.

- **Personal:** estimación del coste de todo el personal que ha intervenido en el proyecto. En este caso, como autor del proyecto, David Castaño Mariscal, como ingeniero.
- **Equipos:** Se estima la amortización del conjunto de equipos utilizados para la realización del proyecto: Pc personal y licencias del software Matlab.
- **Subcontratación:** Apartado en el que se desglosan todas las tareas que requieren de subcontratación a empresas externas. En este caso, todo ha sido realizado sin llevar a cabo esta práctica, por lo que no está incluido.
- **Restos de costes indirectos:** Este apartado recoge los costes generales, dietas, viajes, materiales, imprenta.... Los costes indirectos debido a su dificultad de cálculo se calcula haciendo el 20% del total de material y personal.

En cuanto a los materiales utilizados se considera un ordenador básico, como ha sido el caso del que he utilizado, y la licencia de Matlab. Para obtener los costes imputables al proyecto se aplicaría la siguiente formula de amortización de los mismos.

$(A/B) \cdot C \cdot D$ Dónde:

-A, número de meses desde la fecha de facturación en que el equipo es utilizado.

-B, periodo de depreciación (meses)

-C, costes del equipo.

-D, % del uso que se dedica al proyecto.

Se ha tomado como el periodo de depreciación de los equipos informáticos de 24 meses y en cuanto a la licencia de Matlab de 12 meses. De uso dedicado hemos puesto como referencia un 60% y un 50% respectivamente, por dar unos valores aproximados.

La estimación del tiempo para la elaboración del proyecto sería de:

- Tiempo de desarrollo del proyecto: 8 meses.
- Horas trabajadas/día=1.3 horas
- Horas trabajadas/semana=9.1 horas.

Total de horas empleadas en el proyecto sería de: 300 horas.

A continuación se muestran una tabla resumen con todos los costes empleados.

Personal.					
Nombre y apellidos.	Tarea.	Categoría.	Dedicación (h).	Costes (€/h)	Costes (€)
David Castaño Mariscal.	Documentación	Becario.	120	3.1	372
	Desarrollo y optimización.		280	3.8	1064
				Total.	1436 €

Tabla 6 Presupuesto total personal.

Materiales informáticos y licencias.					
Descripción.	Costes €.	%Uso proyecto.	Dedicación (meses).	Periodo depreciación.	Coste total €.
Ordenador	750	60	8	24	150
Licencia Matlab.	377	50	6	12	92.5
					242.5€

Tabla 7 Presupuesto en material informático y licencias.

Costes indirectos será el 20% del total de material y personal.

Resumen.	Costes €.
Personal.	1436
Material.	242.5
Indirectos.	335
Total.	2014€

Tabla 8 Presupuesto en personal.

Por tanto el presupuesto total del proyecto asciende a la cantidad de dos mil catorce euros.

11. Bibliografía.

1. **Ricardo Lopez Silva, Antonio Lecuona Neumann, Ruben Ventas Garzon, Ciro Vereda Ortiz.** *Impacto computacional del esquema diagonalmente implícito de Runge-Kutta (ESDIRK) en la simulación numérica 1-D transitoria de flujos con cambio de fase.* Leganes : IX CONGRESO NACIONAL DE INGENIERÍA TERMODINÁMICA, 2015.
2. **H., ALEX CARRERA.** <http://www.pmecca.com>. [En línea]
<http://www.pmecca.com/sitio/2014/06/05/dimensionamiento-de-tuberias-de-proceso-en-flujo-bifasico/>.
3. **Giménez, Juan Bautista Emanuel.** *Medición de fracción de vacío basada en técnicas neutrónicas.* Argentina : s.n., 2011.
4. **eng-tips.** *eng-tips.com*. [En línea] <http://www.eng-tips.com/faqs.cfm?fid=1236>.
5. **Ruiz, Sarai Villalana.** *Grupo "O" 6º Semestre SIMULACION DE SISTEMAS.* [En línea] <https://villalana.wordpress.com/1-6-etapas-de-un-proyecto-de-un-modelo-de-simulacion/>.
6. An experimental investigation of true volumetric vapor. [aut. libro] G.G, BARTOLOMEI. *Thermal Engineering*. 1982, págs. 132-135.
7. **Morales, Sergio Ruiz.** <http://apps.morelab.deusto.es>. [En línea]
<http://apps.morelab.deusto.es/teseo/page/thesis/numerical-simulation-of-the-thermal-and-fluid-dynamics-behaviour-of-liquid-vapour-two-phase-flow-in-evaporators-and-condensers>.
8. **B., WALLIS G.** *One-dimensional Two-phase Flow*. s.l. : McGraw Hil, 1969.
9. **S.C., KANDLIKAR.** *Handbook of Phase Change: Boiling and Condensation*. s.l. : Taylor & Francis, 1999.

10. **bibing.** *bibing.us.es*. [En línea]

<http://bibing.us.es/proyectos/abreproy/3718/fichero/Parte+I%252FCapitulo+3.pdf>.

11. **MathsTools.** *MathsTools*. [En línea]

http://mathstools.com/section/main/Metodos_de_Runge_Kutta?lang=es#.

12. **Kandlikar, S. G.** *Handbook of Phase Change: Boiling and Condensation*.

Philadelphia : Taylor y Francis., 1999.

13. **Barragán, Diego Orlando.** *Manual de Interfaz Gráfica de Usuario en Matlab*.

2008.

14. **Pérez, Javier Buhingas.** *Desarrollo de una aplicación en interfaz gráfica de Matlab para la determinación del comportamiento dinámico de un vehículo automóvil*.

Leganés : s.n., 2011.

15. **Gervaso, Mario Salinero.** *Diseño de una banda transportadora mediante guide de matlab*. Leganés. : s.n., 2013.

•

Leganés, a 22 de Junio de 2015

El Ingeniero Projectista

Fdo. David Castaño Mariscal.